

CSE 4084 Multimedia Systems

Homework 1 (Due next week)

For the problems below please prepare a report that contains all the explanations in every step and any intermediate results including images. Place your code at the end of the report.

Problem 1

In this problem and the next, you will implement spatial-domain low-pass filtering using MATLAB, and evaluate the difference between the filtered image and the original image using two quantitative metrics called the Mean Squared Error (MSE) and Peak Signal-to-Noise Ratio (PSNR). Given $N_1 \times N_2$ images $x(n_1, n_2)$ and $y(n_1, n_2)$, the MSE is computed as:

$$MSE = \frac{1}{N_1 N_2} \sum_{n_1=1}^{N_1} \sum_{n_2=1}^{N_2} [x(n_1, n_2) - y(n_1, n_2)]^2$$

The PSNR is defined as $PSNR = 10 \log_{10} \left(\frac{MAX_l^2}{MSE} \right)$, where MAX_l is the maximum possible pixel value of the images. For the 8-bit gray-scale images considered in this problem, $MAX_l = 255$.

Follow the instructions below to finish this problem.

- (1) Download the original image for this question (Lena.gif). The original image is a 256×256 8-bit gray-scale image.
- (2) Convert the original image from type 'uint8' (8-bit integer) to 'double' (real number).
- (3) Create a 3×3 low-pass filter with all the coefficients equal to $1/9$, i.e., create a 3×3 MATLAB array with all elements equal to $1/9$.
- (4) Low-pass filter the original image (converted to type 'double') with the filter created in step (3). This can be done using the built-in MATLAB function "imfilter". The function "imfilter" takes three arguments and returns one output. The first argument is the original image (converted to type 'double'); the second argument is the low-pass filter created in step (3); and the third argument is a string specifying the boundary filtering option. For this problem, use 'replicate' (including the single quotes) for the third argument. The output of the function "imfilter" is the filtered image.
- (5) Compute the PSNR value between the original image (converted to type 'double') and the filtered image by using the formulae given above.

Problem 2

Repeat steps (3) through (5) in the previous question, this time using a 5×5 low-pass filter with all the coefficients equal to $1/25$. Compute the PSNR value up to two decimal points.

Problem 3

In this problem you will get hands-on experience with changing the resolution of an image, i.e., down-sampling and up-sampling. Follow the instructions below to finish this problem.

- (1) Download the original image for this question (original.jpg). The original image is an 8-bit gray-scale image of width 479 and height 359 pixels. Convert the original image from type 'uint8' (8-bit integer) to 'double' (real number).
- (2) In order to avoid aliasing (e.g. jagged edges) when downsampling an image, you will need to first perform low-pass filtering of the original image. For this step create a 3×3 low pass filter with all coefficients $1/9$. Perform low-pass filtering with this filter using the MATLAB function "imfilter" with 'replicate' as third argument.
- (3) Obtain the down-sampled image by removing every other row and column from the filtered image, that is, removing the 2nd, 4th, all the way to the 358th row, and then removing the 2nd, 4th, all the way to the 478th column. The resulting image should be of width 240 and height 180 pixels. This completes the procedure for image down-sampling. In the next steps, you will up-sample this low resolution image to the original resolution via spatial domain processing.
- (4) Create an all-zero MATLAB array of width 479 and height 359. For every odd-valued $i \in [1, 359]$ and odd valued $j \in [1, 479]$, set the value of the newly created array at (i, j) equal to the value of the low-resolution image at $\left(\frac{i+1}{2}, \frac{j+1}{2}\right)$. After this step you have inserted zeros into the low-resolution image.
- (5) Convolve the result obtained from step (4) with a filter with coefficients
[0.25, 0.5, 0.25;
0.5, 1, 0.5;
0.25, 0.5, 0.25]
using the MATLAB function "imfilter". In this step you should only provide "imfilter" with two arguments instead of three, that was the case in step (1). The two arguments are the results from step (4) and the filter specified in this step. This step essentially performs bilinear interpolation to obtain the up-sampled image.
- (6) Compute the PSNR between the upsampled image obtained from step (5) and the original image.

Submission Instructions:

1. Each student should submit his/her own homework. You can discuss the questions with your friends, but you must write your own solutions and code. Group work is not allowed. You can not exchange any written material, code or pseudocode. This also includes material found on the web.
2. Write a detailed report using Word/Latex and convert it to a pdf file. The report must include explanations about each part in each question. You should solve some of the questions by pencil and paper. Then, you can scan your solution and insert it to your report (you can use free Mobile applications such as Adobe Scan).
3. Explain how your scripts and functions work, i.e., which parts of your functions/scripts accomplish which task and how it is accomplished. Include the outputs of your functions and figures to your report. Each figure should have a caption and should be explained in the text.
4. You can use MATLAB or Python for programming assignments. Don't forget to put detailed comments into your functions/scripts to explain what your code is doing. Also indicate the inputs and outputs in the comment section.
5. Combine your report and codes into a single zip file. Plots and figures should go into the report.
6. Name your zip file as "CSE4084_name_surname_hw_no.zip". For example, a student whose name is Ayşe Çalışkan will name her file as: "CSE4084_ayse_caliskan_hw1.zip" for the first homework. Also, write your name, surname and student number as comments at the beginning of your codes.
7. Submit your homework via the class web page at classroom.google.com before the deadline.
8. Late submissions will loose 10 points for each day after the deadline.