E9 241 Digital Image Processing

Assignment 03

Due Date: October 21, 2024 - 11:59 pm **Total Marks:** 30

Instructions:

For all the questions, write your own functions. Use library functions for comparison only.

- Your function should take the specified parameters as inputs and output the specified results.
- Also provide the wrapper/demo code to run your functions. Your code should be self-contained, i.e., one should be able to run your code as is without any modifications.
- For python, if you use any libraries other than `numpy`, `scipy`, `scikit-image`, `OpenCV`, `pillow`, `matplotlib`, `pandas` and default modules, please specify the library that needs to be installed to run your code.
- Along with your code, also submit a PDF with all the results and inferences. Include answers to subjective questions, if any.
- Put all your files into a single zip file and submit the zip file. Name the zip file with your name.
- **Vectorize your code. Non-optimized code may be penalized.**

Frequency Domain Filtering:

1. Pass the image `dynamicCheckerBoard.png` through an ideal low pass filter (ILPF), ideal bandpass filters (IBPFs), and an ideal high pass filter (IHPF).

   - The expression for the ILPF is

   $$H_{\text{ILPF}}(u, v; D_0) = \begin{cases} 1 & D(u, v) \leq D_0 \\ 0 & D(u, v) > D_0 \end{cases}$$

   where D_0 is a positive constant referred to as the cut-off frequency and $D(u, v)$ is the distance between a point (u, v) in the frequency domain and the center of the frequency rectangle, i.e., $D(u, v) = \sqrt{(u - P/2)^2 + (v - Q/2)^2}$, where P and Q are the number of rows and columns in the image.

   - An IHPF with a cut-off frequency at D_0 can be computed as

   $$H_{\text{IHPF}}(u, v; D_0) = 1 - H_{\text{ILPF}}(u, v; D_0)$$

   - An IBPF with a lower cut-off at D_l and a higher cut-off at D_h can be computed as a product of ILPF and IHPF.

   $$H_{\text{IBPF}}(u, v; D_l, D_h) = H_{\text{ILPF}}(u, v; D_h) \times H_{\text{IHPF}}(u, v; D_l)$$

   Perform different frequency domain filters on the DFT of the input image and find their IDFT to get the filtered outputs. Use the following parameters for each filter:

Filter	Function
Low pass	ILPF(10)
Band pass - 1	IBPF(10, 20)
Band pass - 2	IBPF(20, 30)
High pass	IHPF(30)

Plot the filter and the filtered images. What do you observe in the resultant images? How do they relate to the filters?

Note: The filters given above are centered in the frequency domain. To use such centered filters, you will either need to shift the filter to $(0, 0)$ (by using fftshift in MATLAB or the corresponding function in python) or center the DFT of the image. To center the DFT of the image, you can either shift the DFT of the image or scale each image pixel $I(x, y)$ by -1^{x+y} before computing its DFT. If you center the DFT of the image, then you will need to compensate for it by multiplying the image obtained from the inverse DFT of the filtered image by -1^{x+y}.

2. Filter the image `characters.tif` in the frequency domain using an ILPF and the Gaussian low pass filter (GLPF) given by

$$H_{\text{GLPF}}(u, v; D_0) = exp(-D^2(u, v)/2D_0^2)$$

where all the terms are as explained in the last part. For $D_0 = 100$, compare the results. Do you observe any artifacts?

(20+10 Marks)