Middle East Technical University Department of Computer Engineering

CENG 466

Fundamentals of Image Processing

Fall '2020-2021

Take Home Exam 2

Due date: December 25 2020, 17:00

1 Objectives

The purpose of this assignment is to familiarize you with the fundamental frequency domain image enhancement techniques. You are given several tasks. You are expected to carry out these tasks using algorithms in lecture notes and the text book.

2 Specifications

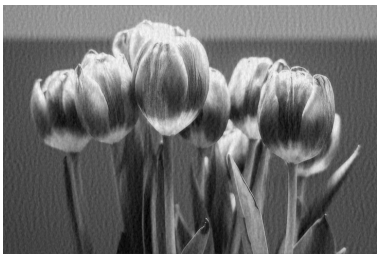
You are given three questions, which you should solve with your own algorithms. In addition to the solutions, you are required to prepare a report that explains your methodology and includes the analysis of the results and your comments on them. The report should be **3-5 pages** long and should be prepared in IEEE Conferance Proceedings Template (LaTeXis recommended) provided in the following link.

`https://www.ieee.org/conferences_events/conferences/publishing/templates.html`

- Grading will be based on the quality of the outputs, script contents and the report
- The report should clearly explain the methodology and rationale behind the algorithm design. It should also explain the difficulties encountered in the design, implementation and experimentation stages, and your solutions on them. Last but not least, the report should contain your comments on the results. Even if the results does not match your expectations you should discuss the encountered situation.

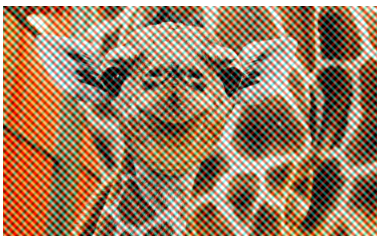
2.1 Question 1 (30 Points) - Noise Reduction using Frequency Domain Filtering

In this part, you are required to identify the type of noise in the images A1.png, A2.png and A3.png shown in Figure 1. Your task is to investigate the present noise and remove them using Frequency domain filtering to recover the original image. While removing the noise you should try to preserve informative structures like edges and boundaries as much as possible. Note that full recovery is not possible in some cases due to the information loss. Implement your solution as a MATLAB or Python script named **the2_part1** which should remove the noise in the given images and save the output image to the same folder with the name **AX_denoised.png**.
You can use fft2(), ifft2() and fftshift() functions.



(a) A1

(b) A2



(c) A3

Figure 1: Noisy images of part 1

2.2 Question 2 (20 Points) - Edge Detection

In this part you are given two images B1.jpg, B2.jpg and B3.jpg shown in Figure 3. Your job is to find edge maps of these images in Fourier domain. Implement your solution in a MATLAB or Python script named **the2_part3**. This script should save the edge maps into the same folder with the name **BX_output.png**.

2.3 Question 3 (50 Points) - Image Compression with Wavelet Decomposition

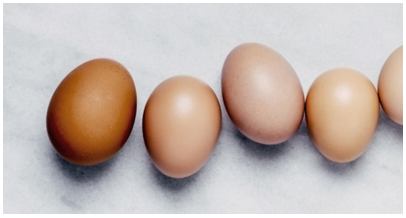
Image compression is an important problem in image processing. There are two kinds of compression algorithms, namely, lossless and lossy compressions. As their name indicates, images compressed with lossless methods can be decompressed without loss of any information, while an image, compressed with a lossy method is decompressed with some information loses. However, the information losses can be tolerated most of the time and cannot be perceived by the end user. Lossy methods usually achieves higher compression rates than lossless methods and consequently they are used in practice. One of the most popular lossy compression standard is JPEG standard which uses discrete cosine transformations. Suppose that, you are asked to develop and implement an image compression algorithm based on the wavelet decomposition.. Your task is divided into three steps:

- Design an image compression algorithm based on wavelet decomposition. Implement a compression and a decompression methods for the algorithm, named compress and decompress, respectively. Both methods will take two arguments, namely, path to input file and path to output file. You can add any number of optional arguments to compress method. However, decompress method can use only the information stored by compress method. Your algorithm is expected to handle gray-level images stored as a single 2D matrix, other images will not be tested. You are allowed to use dwt2 and idwt2 methods from wavelet toolbox. However, you cannot use any other wavelet toolboxes during implementation.

- You are given five images (C1.png, C2.png, C3.png, C4.png, C5.png) to evaluate your implementation. Please, write a MATLAB or Python script named **the2_part3**. Compress/decompress the given images using your image compression algorithm and JPEG standards, for different parameter

(a) B1

(b) B2



(c) B3

Figure 2: Images of part 2

sets including quality for JPEG standards. For each parameter set and image, print the following statistical information to console, also explain these findings in your report in detail.
Compression ratio, calculated with the following formula;

$$\text{size of the original image / size of the compressed file} \tag{1}$$

Mean square error (MSE), calculated with the following formula;

$$MSE = \frac{1}{MN} \sum_{y=1}^{M} \sum_{x=1}^{N} [I(x, y - R(x, y))]^2 \tag{2}$$

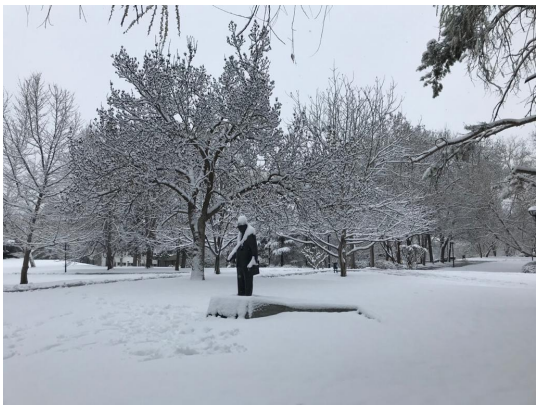
where original image is I whose size is NxM and R is the decompressed image.

You may format the output as you wish, as long as it is easy to understand.
Hint: Built-in imread and imwrite methods support decompression/compression with JPEG standards in MATLAB.

- Please, explain your compression algorithm in detail and compare your algorithm to JPEG standards using statistical information computed during step 2.

3 Regulations

1. **Group:** You are required to do your assignment in a group of two. If there is an unclear part in your code, we may ask any of the group member to describe that code segment. Also group members may get **different** grades. We reserve the right to evaluate some or all of the groups to determine the contribution of each group member to the assignment.



(a) C1

(b) C2

(c) C3

Figure 3: Images of part 3

2. **Programming Language:** You must code your program in MATLAB or Python. You are expected make sure your code runs successfully on department lab machines.
3. **Late Submission:** Late Submission is **not** allowed!
4. **Newsgroup:** You must follow the discussion group on Odtuclass for discussions and possible updates on a daily basis.

4 Submission

Submission will be done via Odtuclass. Create a tar.gz file named THEX.tar.gz that contains all your source code files and the report as a PDF file. Do not send the input and output images. Only one member should submit the homework. Hence, do not forget to **write your names and student id's at the beginning of the scripts.**

5 Cheating

We have zero tolerance policy for cheating. People involved in cheating will be punished according to the university regulations.

Cheating Policy: Students/Groups may discuss the concepts among themselves or with the instructor or the asistants. However, when it comes to doing the actual work, it must be done by the student/group alone. As soon as you start to write your solution or type it, you should work alone. In other words, if you are copying text directly from someone else - whether copying files or typing from someone else's nots or typing while they dictate - then you are cheating (committing plagiarism, to be more exact). This is true regardless of whether the source is a classmate, a former student, a website, a

program listing found in the thrash, or whatever. Furthermore, plagiarism even on a small part of the program is cheating. Also, starting out with code that you did not write, and modifying it to look like your own is cheating. Aiding someone else's cheating also constitutes cheating. Leaving your program in plain sight or leaving your computer without logging out, thereby leaving your programs open to copying, may constitute cheating depending upon the circumstances. Consequently, you should always take care to prevent others from copying your programs, as it certainly leaves you open to accusations of cheating. We have automated tools to determine cheating. Both parties involved in cheating will be subject to disciplinary action. [Adapted from http://www.seas.upenn.edu/cis330/main.html]