Amirkabir University of Technology (Tehran Polytechnic)
Digital Image Processing – Spring 2020
Assignment 4
Mastering Image Restoration Techniques

Please remember:

1. What you must hand in includes the assignment report (.pdf), source codes (.m) and output files (.png). Please insert each part in a different folder, and zip them all together into an archive file named according to the following template: HW4_XXXXXXXX.zip Where XXXXXXXX must be replaced with your student ID.

2. Your work will be evaluated mostly by the quality of your report. Don't forget to explain what you have done, and provide enough discussions which proves you have realized the subject.

3. 5 points of each homework belongs to compactness, expressiveness and neatness of your codes and report.

4. By default, we assume you implement your codes in MATLAB. If you're using Python, you have to use equivalent functions when it is asked to use specific MATLAB functions.

5. Your codes must be separated for each question, and for each part. For example, you have to create a separate .m file for part b. of question 3. Please name it like p3b.m.

6. "Keywords" will help you find useful information about the problem. They may also include some ideas for solving that problem.

7. Using built-in functions is not allowed, except for simple operations like reading, displaying, converting and saving images, or in cases it is clearly mentioned in "Allowed MATLAB Functions" section of each problem.

8. Please upload your work in Moodle, before deadline.

9. If there is any question, please don't hesitate to contact us through the following email addresses: ali.the.special@gmail.com, dalirani@aut.ac.ir

10. Unfortunately, it is quite easy to detect copy-pasted or even structurally similar works, no matter being copied from another student or internet sources. Try to send me your own work, without being worried about the grade! ;)

1. Applying Spatial Filtering on a Binary Image (10 Pts.)

Keywords: Spatial Filtering, Arithmetic Mean Filtering, Geometric Mean Filter, Harmonic Mean Filter, Order-Statistic Filter, Midpoint Filter, Median Filter, Min Filter, Max Filtering

Spatial Filtering is a powerful mean to modify or enhance an image. As a neighborhood operation, it works by applying some algorithms to the values of the pixels in the neighborhood of any given pixel, to determine its value in the output image.

In this problem, you are to work with some basic spatial filters which work similar to most of the filters you are already familiar with. After understanding how a particular filter affects the image, please give a brief verbal description of the result.

Read Input Image as a gray scale image.

(a) Arithmetic mean filter with the size of 3×3 and 7×7.
(b) Geometric mean filter with the size of 3×3 and 7×7.
(c) Harmonic mean filter with the size of 3×3 and 7×7.
(d) Contraharmonic mean filter with the size of 3×3 and 7×7, where Q = 1.5 and Q = -1.5.
(e) Midpoint filter with the size of 3×3, 7×7 and 9×9.
(f) Median filter with the size of 3×3, 7×7 and 9×9.
(g) Min filter with the size of 3×3 and 9×9.
(h) Max filter with the size of 3×3 and 9×9.

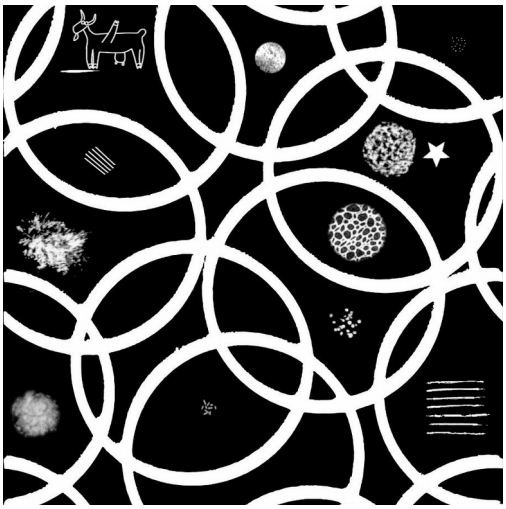


Image 1-1.png

Allowed MATLAB functions: im2double() , rand(), random() ,log() , exp() , find()

2. Adding Different Type of Degradation to Images (15 Pts.)

Keywords: Spatial Filtering, Gaussian Noise, Salt-and-Pepper Noise, Uniform Noise, Rayleigh Noise, Gamma Noise, Exponential Noise, Atmospheric Turbulence, Motion Blur

Although the ultimate goal of Image Restoration is to achieve a better image quality, sometimes it is more desirable to add degradation to the images on purpose. It is mostly in the cases when a more natural and realistic image is preferred, or the objective is to build creative effects. Also, sometimes it is used to create test data in order to evaluate restoration methods.

In this problem, you are going to use known models of degradation to degrade images. It would be beneficial to see the effects of each degradation and their parameters on an image.

In this problem, you are going to use known models of degradation to degrade images. It would be beneficial to see the effects of each degradation and their parameters on an image. Therefore you will be able more easily to identify which restoration methods you need to use in order to restore corrupted images.

(a) Implement a function to add Gaussian noise to an image. Your function should be able to specify the noise specific parameters. Apply it on the image "2-1.png" with three arbitrary settings, and display the results and compare histograms of clean and noisy images. Read image 2-1.png as a gray scale image.
(b) Repeat part a. for salt-and-pepper noise.
(c) Repeat part a. for uniform noise.
(d) Repeat part a. for Rayleigh noise.
(e) Repeat part a. for gamma noise.
(f) Repeat part a. for exponential noise.
(g) Implement a function to add atmospheric turbulence to an image. Your function should be able to specify the parameter k, which determines the degree of turbulence. Apply it on the image "2-2.png" with three settings to produce "low", "mild" and "severe" atmospheric turbulence, and display the results.
(h) Implement a function to add motion blur to an image. Your function should be able to specify the parameters a and b, which determine the distance of the motion in each directions. Apply it on the image "2-3.png" with three arbitrary settings, and display the results.

2-1.png, Lonely Tree, contains several intensity

2-2.png Naghshe Rostam

2-3.png, Nader Shah Afshar

Note: The results in part g. and h. must be in RGB space.
Allowed MATLAB functions: imhist() , exp(), fftshift() ,fft2() , real() , meshgrid()

3. Noise Reduction (20 Pts.)

Keywords: Image Denoising, Spatial Filtering, Additive Noise, Gaussian Noise, Impulse (Salt-and-Pepper) Noise, Periodic Noise

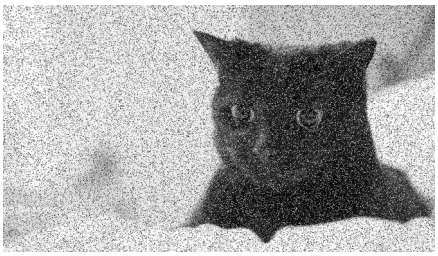
Spatial Filtering is widely used in image restoration, especially when the goal is to reduce noises of type additive. Choosing appropriate filter depends on many factors, including noise type, amount and distribution. The same goes for the parameters associated with it, like filter size or filter specific constants.

The goal of this problem is to demonstrate the usage of noise reduction techniques in image restoration, as well as testing your skills in finding the most convenient filters and filter settings for noise reduction purpose. Although the emphasis is on spatial denoising techniques, in some cases you may find frequency domain filtering useful as well.

(a) Load the image 3-1.png. Depending on the characteristics of the noise you observe, select an appropriate filter as well as suitable parameters to reduce the noise in the image. You are free to apply more than one filter as well.
(b) Repeat part a. for the image 3-2.png.
(c) Repeat part a. for the image 3-3.png.
(d) Repeat part a. for the image 3-4.png.
(e) Repeat part a. for the image 3-5.png.

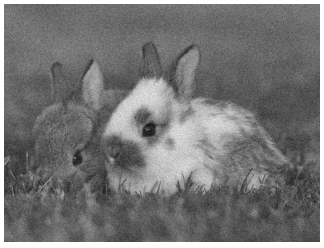
3-1.png, Abadan Hospital

3-4.png, baby



3-2.png A cat from land of Mordor

3-5.png, Qajar



3-3.png, Rabits

Note: You may not be able to remove noises entirely, but your final result must be worth the effort.
Allowed MATLAB functions: imfilter() , fspecial() , roifilt2() , imgaussfilt() , wiener2() , medfilt2() , ordfilt2() , imboxfilt() , fft2() , ifft2() , fftshift() , abs() , angle() , log() , im2double()

4. Comparison of Different Methods for Resolution Enhancement (15 Pts.)

Keywords: Image Interpolation, Image Super-Resolution, Nearest-Neighbor Interpolation, Bilinear Interpolation, Nearest Neighbor Value Interpolation, Non-uniform Interpolation

Image Interpolation is an image processing method by which the number of pixels comprising an image is increased. Nearest-Neighbor Interpolation is the simplest way of image interpolation, where the values of an unknown pixel is determined by the value of its nearest pixel, while Bilinear Interpolation works by interpolating pixel intensity value using more pixel neighbors, based on their distances to the unknown pixel.

The purpose of this problem is to practice enhancing image resolution using these methods. You are going to work with image 4-1.png, as the input image.

(a) Rescale the input image by the factor of 2 using nearest-neighbor interpolation method.
(b) Rescale the input image by the factor of 2 using bilinear interpolation method.
(c) Rescale the input image by the factor of 2 using nearest neighbor value interpolation. Note that in this method, the missing pixel value is estimated by the nearest value rather than the distance. In other words, the nearest neighbor value interpolation considers the four pixels directly surrounding the empty location, and selects the one whose value is almost equal to the value obtained by bilinear interpolation method.

4-1.png, Recreation on a famous painting

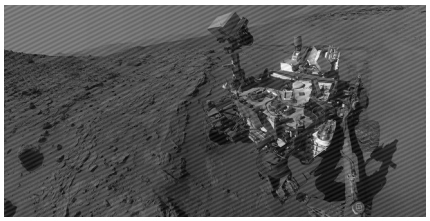
Allowed MATLAB functions: size() , ceil() , meshgrid() , psnr()

5. Applying Notch Filters to Remove Periodic Patterns (25 Pts.)

Keywords: Notch Filters, Periodic Noise

Notch Filters are used to reduce the effect of periodic noise or repetitive patterns from an image. They do so by attenuating a selected frequency – and some of its neighbors – and leave other frequencies of the image Fourier transform unchanged. In this problem, you will get familiar with notch filters and their capabilities in removing periodic patterns – including noises – from an image.

(a) Load the image 5-1.png and display the corresponding spectrum. As can be seen, the image suffers from a periodic noise. Use notch filters to remove it from the image, and display the result.
(b) Repeat part a. with the image 5-2.png.
(c) Load the image 5-3.png and display the corresponding spectrum. As you can see, the blinds have obscured the view. Because of its natural pattern, one can use image restoration techniques in frequency domain to deal with this problem. Design and apply a notch filter to reduce the effect of this periodic pattern from the image, and display the results.
(d) Repeat part c. with the image 5-4.png.
(e) Repeat part c. with the image 5-5.png.

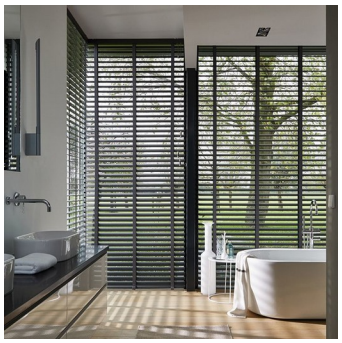


5-1.png Mars Selfie

5-2.png, Path of robot on Mars surface

5-4.png Room

5-5.png skyscraper



5-3.png luxury house

Note: remember that the input images in part c, d and e are in RGB space.
Allowed MATLAB functions: imtool() , mat2gray() , fft2() , fftshift() ,

6. Some Explanatory Questions (15 Pts.)

Please answer the following questions as clear as possible:

(a) What is the difference between additive noise and multiplicative noise? Which one does “salt-and-pepper” noise belong to?
(b) Imagine you want to remove a periodic noise from an image, and you are not allowed to use frequency domain filtering. What type of filter are you going to use? How does the frequency of the noise affect your kernel size?
(c) Imagine you want to remove a Gaussian noise from an image, and you are not allowed to use spatial domain filtering. What type of filter are you going to use? How does the mean and variance of the noise affect your filter parameters?
(d) Why does the last column of an affine transformation matrix consist of two ‘0’s and a ‘1’?
(e) Write an appropriate transformation matrix for a transformation consisting of a horizontally scaling of 2x, a counter-clockwise rotation by 60 degrees and a vertically translation by 20 pixels.

Good Luck!
Farhad Dalirani
Ali Abbasi