University of Tehran

School of Electrical and Computer Engineering

Digital Image Processing

Instructor: Hamid Soltanian-Zadeh

Assignment 1

Due Date: 10th of Esfand 1403

Corresponding TA:

Zahra Nakhaei

Zahranahaei1999@yahoo.com

Spring 2025

Table of Contents

Problem 1

Here, we have an example to get familiar with MATLAB for Digital Image Processing.

a. Use imread function to read image “lenna_rgb.png.”
b. Use imshow function to show the image.
c. Now, convert the image to grayscale using rgb2gray function and display the result.
d. Convert the gray image you made in step “c” into double using im2double function and discuss the differences.
e. Save the image you made in step “d” in “.jpg” format.
f. Magnify and display the gray image by scales 5, 1/2, and 1/4, then discuss the effect of reducing spatial resolution.
g. Like Figure 2.16 in the textbook, try to plot the scan line of 150th line of the gray image.

Problem 2

Now load "skull.tif" image. As you see, this is a 256-intensity level image. Display the image in 64, 16, 4, and 2 intensity levels while keeping the image size constant. Discuss the effect of varying the number of intensity levels in a digital image.

Problem 3

As mentioned in the textbook, geometric spatial transformations and image registration are necessary steps for some image processing applications. In this problem, we will try to get to know some transformations used in this field.

Load the image “T.tif” and apply all the transformations such as Scaling, Rotation, Translation, and Shear to it. You can use the Affine Matrices mentioned in Table 2.2 of the textbook. Set the parameters of each transformation by your own idea and report them clearly for each part. Display the results.

Note: You **cannot use** MATLAB built-in functions such as *imresize, imtranslate, imrotate,* and *imshear* for this problem.

Problem 4

In this section, we want to implement some important logical operations.

a) Create two arrays of zeroes 200 × 200.
b) In the first array, add a rectangle of amplitude 1 starting from (20,30) with a length of 80 and a height of 100.
c) In the second array, add a rectangle of amplitude 1 starting from (40,70) with a length of 110 and a height of 90.
d) Write functions to implement logical operations brought in Section 2.6.4 in the textbook
e) (NOT, AND, OR, AND-NOT, XOR), just like Figure 2.33. The functions must get the name of each operation. Display your results like Figure 2.33.

Problem 5

Local averaging is a simple, intuitive, and easy way for smoothing images and is often used to reduce noise. The idea of local averaging is to replace each pixel value in an image with the mean value of its neighbors, including itself, like Figure 2.35 of the textbook.

a. Write your own code to do local averaging on "lenna.png" with a window size of 3×3.

b. Repeat "Part a" with different window sizes such as 7×7, 11×11, and 23×23. Discuss the effect of window size on local averaging.

Problem 6

Interpolation is the process of using known data to estimate values at unknown locations.
In this problem, we want to compare three different interpolation approaches used for image shrinking.

a) Load the image "watch.tif."
b) Shrink the images by a factor of 5 (you can use imresize function) using nearest, bilinear, and bicubic interpolation techniques. Zoom the reduced image back to its original size and discuss their differences in terms of details.

Problem 7

a) load ‘WashingtonDC.tif’ and set the 6th bit value of each pixel to zero. Compare the result image with the original one and display the difference. How much Information is lost?

b) The "angiography_mask.tif" image is a mask X-ray of the top of a patient's head before injection of iodine into the bloodstream. The "angiography_live.tif" image is a sample of a live image taken after the injection. Use subtraction to highlight the blood vessels. To enhance visualization, you can also display the negative of the subtraction result .

Descriptive Assignments

Please solve the following questions of the 2nd Chapter of the textbook (Digital Image Processing – Edition 3): 5, 7, 9, 10, 19, 21, 23.

General Rules

1. Put written codes for each question in one m-file, and for each section, intercept them by %%.
2. Analytical problems can be solved on papers, and there is no need to type the answers. The only thing that matters is the quality of your pictures. Scanning your answer sheets is recommended.
3. Simulation problems need report as well as source code and results. This report must be prepared as a standard scientific report.
4. Your report is particularly important in the correction process. Please mention all the notes and assumptions you made for solving problems in your report.
5. You have to prepare your final report, including the analytical problems answer sheets and your simulation report in a single pdf file.
6. Finalized report and your source codes must be uploaded to the course page as a ".zip" or ".rar" file with the file name format as:
   DIP_HW#_FirstName_Surname_StudentNumber.zip
7. Plagiarisms will be strictly penalized.
8. While AI tools like ChatGPT can be helpful for brainstorming and generating ideas, they are not a substitute for understanding and writing your own code. Directly copying code from AI tools without attribution is considered plagiarism and will result in penalties.
9. You are allowed a total grace period of 150 hours to submit late assignments for all of your exercises.
10. You may ask your questions from the corresponding TA of each assignment.