Homework #1:
Introduction to Image Processing in MatLab/Octave

Assigned: 04.03.2020 Due: 20.03.2020

1. Objective

The purpose of this initial homework is to make you ready for the upcoming assignments. In the assignments, you will use Matlab with Image Processing Toolbox[1] and/or Python. But this is not a Matlab/Python programming course and you will not learn programming in Matlab/Python in the lectures, except a brief introductory tutorial. So, **it is your own responsibility to learn Matlab and Python programming**.

2. Preparation

Task 1 - Installation

1. Obtain and install Matlab with Image Processing Toolbox on your PC. The version of the Matlab does not matter but in your assignments, indicate which version you use (recommended: 2014 or later versions). You can use GNU Octave[2] instead of Matlab, but in your assignments, indicate which version you use.

Task 2 – Matlab/Python Programming

If you are not familiar with Matlab and Python programming, try learning how to program in Matlab and Python, using online tutorials and material provided in Moodle. You should learn at least scalar, vector, and matrix operations, basic loops and control structures, plotting, and functions.

3. Assignment

Task 3 – Plotting in Matlab/Octave

Write a Matlab/Octave script to generate the plots in Figure 1. The first plot in the figure shows $y = e^x$ function. In the second plot, there are two functions:

$$y = e^x * \sin x^2$$

$$y = e^x * \cos x^2$$

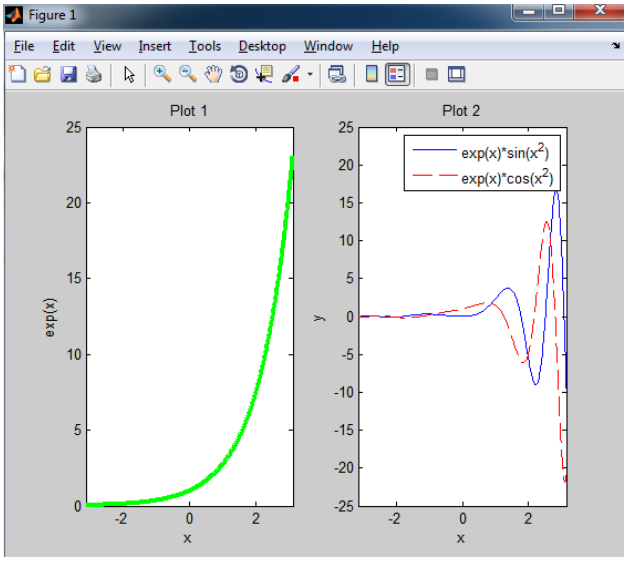

Figure 1 – Output of Task 3.

[1] https://www.mathworks.com/products/image.html?BB=1
[2] https://www.gnu.org/software/octave/

Notes:

1. For all 3 functions, x values should be defined in the range of $-\pi\ to + \pi$.
2. You need to use *subplot* function to display multiple plots side by side.
3. Generate the plot titles, axis labels, axis ranges, legends, line styles and line widths as shown in the figure.

Task 4 – Basic Image Processing in Matlab/Octave

To get started with digital image processing in Matlab/Octave, apply the functions in this tutorial step by step. Then write a Matlab/Octave script to generate the figures in Figure 2. Again, you will need *subplot* function. There are four images in Figure 2:

- original RGB image (top-left)
- grayscale version of the original image (top-right)
- binary version of the original image (bottom-left)
- histogram of the grayscale image (bottom-right)

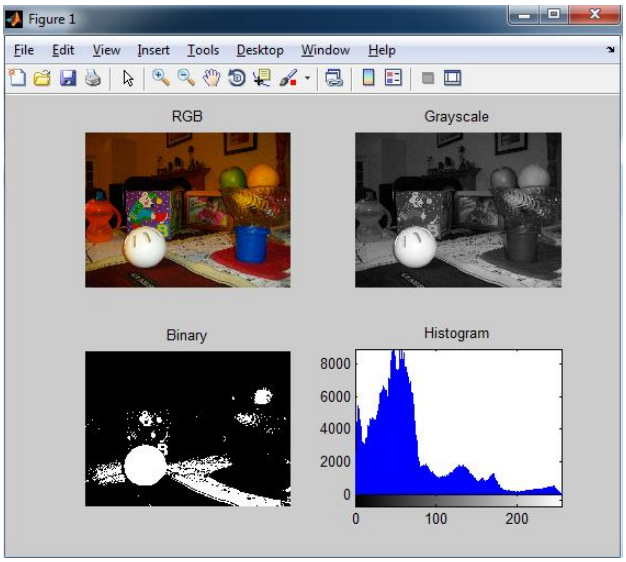

Figure 2 – Output of Task 4.

(**Hint:** You can type **"help imdata"** on Matlab command line to see the list of sample images provided in your Matlab toolbox. The image in this example is '*toysflash.png*'. You can use any other color image.)

4. Submission

- This homework will be done individually.
- For Task 1 and Task 2, you don't have to submit anything.
- Prepare a **README.txt** file including your *name*, *surname*, *student ID*, and *your Matlab/Octave versions*.
- For Task 3 and 4, you should submit your Matlab/Octave source codes in different files called **HW1_Task3.m** and **HW1_Task4.m,** respectively. At the top of your .m files, write your name, surname, and student ID as a comment.
- Your submission directory will include the following files:
  - README.txt
  - HW1_Task3.m
  - HW1_Task4.m
- Place all your files in a zip archive with name **HW1_StudentID_Surname_Name.zip** and submit through the Moodle submission module.
- If you have further questions, you can send me an e-mail or come to my office.

5. Late Submission Policy

Deadline for homework submissions is **23:59 pm** at the specified date. For each additional day, **25% cut-off** will be applied.

6. Grading

Grading will be done according to the following scheme:

- Task 3 (50 pts)
- Task 4 (50 pts)

Dr. Zeynep ÇİPİLOĞLU YILDIZ