MATLAB/OCTAVE
PROGRAMMING ASSIGNMENT

In this programming assignment, you will implement 4 functions in Octave or Matlab programming language. Matlab is licenced, Octave is open-source (free). You can choose either of two. It is recommended to use Octave.

If you choose Matlab, you need to have image processing toolbox installed. If you choose Octave, you need to install <<image>> package. To install image package follow steps below:

1. Download image package at: https://octave.sourceforge.io/image/index.html

2. Install package using "pkg install <fullapath2downloaded_file_in_step_1>". It may take 2-3 minutes to install.

3. To check if it is installed, you can run "pkg list".

```
pkg list
-| Package Name  | Version | Installation directory
-| --------------+---------+------------------------
-|        image *|   1.0.0 | /home/jwe/octave/image-1.0.0
```

4. To use the package, you should write "pkg load image". Now all image library functions are ready to use.

Homework contains mainly 4 parts. You need to fill required places inside the function. **You have to write your code, do not use ready functions**. If your implementation is correct, you get corresponding points. **You should not change test function**. It will test your code and prints final grade. Here are details of the functions:

1. (30p) Convolution: function convolve_im(im, filter).

You need to implement your own convolution filter. Convolution operation is summarized in the Figure-1. The filter size is (3x3) and image size (7x7) for the figure. The corresponding result will be 6x6 (imageSize-filterSize/2) if no padding applied (remember from the lecture). In the homework, **image and the filter size can be different** (i.e. filter size 5x5, image size 100x100, result size 98x98). You can also see animation at: https://i.stack.imgur.com/SFST9.gif

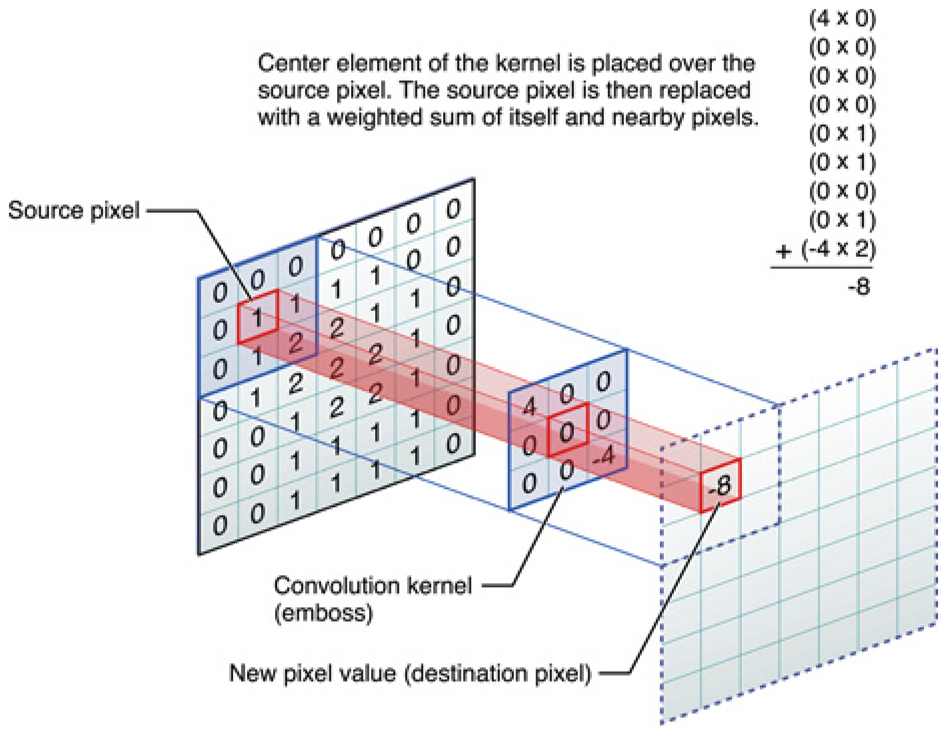
Center element of the kernel is placed over the source pixel. The source pixel is then replaced with a weighted sum of itself and nearby pixels.
(4 x 0)
(0 x 0)
(0 x 0)
(0 x 0)
(0 x 1)
(0 x 1)
(0 x 0)
(0 x 1)
+ (-4 x 2)
-8
Source pixel
Convolution kernel (emboss)
New pixel value (destination pixel)

Figure-1. Convolution operation

The filter size always be odd number (1, 3, 5, 7, 9 …) and image will be gray scale (single channel). Thus, you do not need to worry about colored images which contains 3 channels.

2. (25p) Histogram: function image_histogram(img)

In this part of the work, you need to find histogram of an gray scale image (single channel). The histogram is an array for showing the number of pixels in an image at each different intensity value found in that image. Briefly, it tells how many pixel value p_i in the image. You do not need to plot histogram, only find it. The result should be the same as imhist(image) function but you have to implement it on your own. **Do not use functions.**

3. (25p) Find Eigen values: function eigen_val(mat1):

You need to find eigen values with your own code. You can read following text to get idea about what is eigen value: http://mathworld.wolfram.com/Eigenvalue.html

You should not use ready functions. Your function should **return only the eigen values, it should not return eigen vectors.**

4. (20p) Mathematical function: perform_math(a,b,c):

Suppose that a is 3x3, b is 3x1 and c is 1x3. Implement following function:

$$\mathbf{f(a,b,c) = b^2 + a*c + exp(c^2 - b)}$$

You need to be careful about the dimension. If dimension are mismatch, think the operation is element-wise.

NOTES: **Please be sure that dimensions should be OK for each part** otherwise program will terminate unexpectedly. It is your responsibility to make it correct. If program does not run, you get 0 (zero) from the homework. Please send your homework through UZEM with following name convention: id_nameSurname.zip

Please do not copy code from any of your college. If source codes are the same with your college, you get -50 points from the homework. Do your homework alone. Do not say something like “we studied together”. This is not an excuse.

Deadline: 15 April 2018 at 23:59.

Late submissions will be punished -25 point for each day.