Universidade de São Paulo
Teaching Assistant: Sherlon Almeida Email: sherlon@usp.br
Teaching Assistant: Bruno Resende Email: messias@ifsc.usp.br

Try to code the assignment by yourself. Plagiarism is not tolerated

Assignment 4
Image restoration

Problem Statement

In this assignment you have to implement functions in order to enhance and/or filter images using a series of operations. Read the instructions for each step. Use python with the **numpy**, **imageio** and **scipy** libraries.

Your program must have the following steps:

1. **Parameters input:**
   a. Filename for the input image (**I**),
   b. Choice of method:
      1 - Constrained least-squares Filter
      2 - Richardson-Lucy
   c. Parameters for the methods:
      i. **Method 1 - CLSQ:**
         1. **k:** the size of the degradation filter
         2. **sigma:** standard deviation for the gaussian degradation (blurring)
         3. **gamma** parameter [0, 1)
      ii. **Method 2 - RL:**
         1. angle related to the PSF
         2. number of steps
         *num_pixel_dist = 20 (please, use this for the psf filter)
2. **Compare the restored image with the original one.**

Constrained least-squares for digital image restoration

Linear least-squares problems appear everywhere in statistics and computer science. Constrained least-squares is a least-square problem with restrictions imposed on the parameters to be estimated. In our case, we start with the assumption that our degraded image is produced by the following operation:

Degraded image = Degradation Function x Original Image + Noise

$$g = hf + \eta$$

The restriction (constraint) will be

$$||g - hf||^2 = ||\eta||^2$$

Thus we just need to find a f which solves that. In the Fourier space, the regularized solution for this problem at the pixel position (u, v) will be

$$F(u,v) = \left(\frac{H^\star}{|H|^2 + \gamma|P|^2} G\right)(u,v)$$

In the above equation the gamma (γ) is the regularization parameter and P is the Fourier transform of the Laplacian operator. Your task will be to perform a gaussian degradation in an image and after that try to restore the original image using the above filter. Your output should be a comparison with the original image.

Note: You should perform a clip operation right after the CLSQ application to remove artifacts:

```
estimated_img = np.clip(
      estimated_img.astype(int), min_val_degraded, max_val_degraded)
```

Linear motion blur

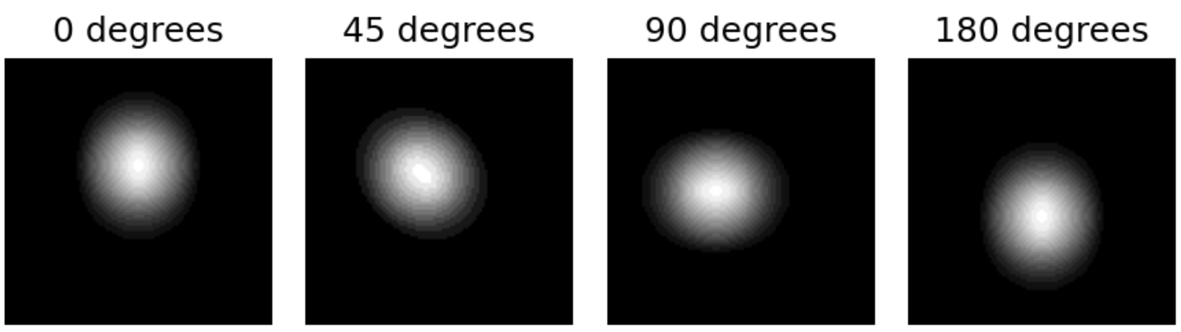
One of the most common causes of image degradation is motion blur. It's caused by the fact that any equipment will not record a single instant of time of a scene. But an interval of the time, that is, the exposure time.

Several methods have been proposed to mitigate linear motion blur effects. Here, you will try to implement one of the first methods to address this motion blur degradation. The Richardson-Lucy deconvolution is an iterative algorithm that can be used to perform a deblurring task in a given image. To use the original algorithm proposed in the nineties is necessary to know or at least have a good bid of the point-spread function that created the blur effect. Although, novel works have been proposed to address this condition.

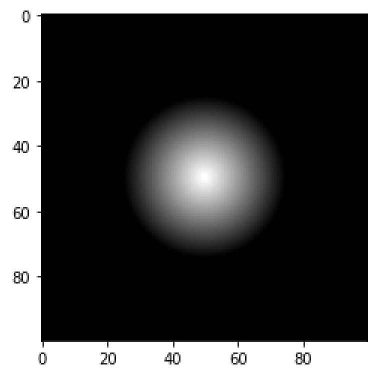
In this task, you must implement the Richardson-Lucy deconvolution and apply a list of predetermined images to it. Please take a look at the following colab notebook that we have made to guide you in this task.

https://drive.google.com/file/d/1yZt9SAqkxhTUjlf4Pk99YPMyLpiwhFwc/view?usp=sharing

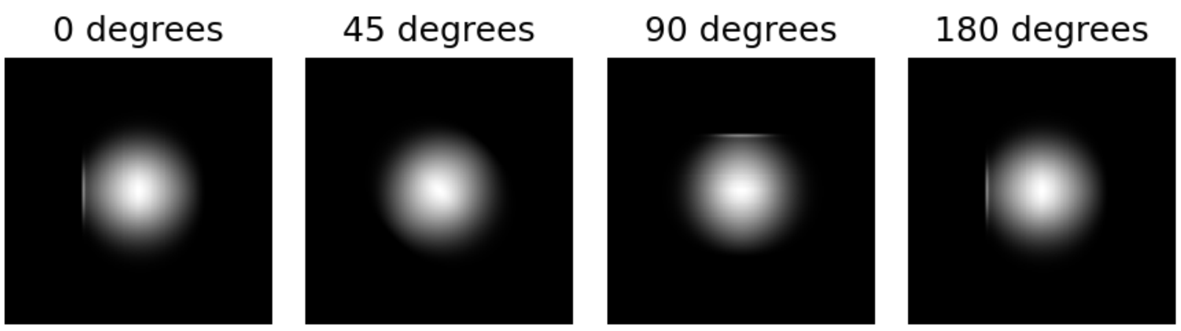
The following images show the degradation effect created by a point spread function with different angles



Below, is the original image:



Your implementation of the Richardson-Lucy algorithm should result in a output similar to this:



Notice that besides some artifacts we have a good restoration of the original image.

Comparison with original image

After generating the output image $\hat{\mathbf{I}}$, compare it with the original image $\mathbf{I}$ using RMSE. Since $\mathbf{I}$ has values between 0 and 255, you must normalize $\hat{\mathbf{I}}$ so that it also has values in the same range in the **uint8** format.

$$\text{RMSE} = \sqrt{\frac{1}{MN}\Sigma\Sigma(I(i,j) \; - \; \hat{I}\,(i,j))^2}$$

Where M x N is the size of the image.

Input and Output

Examples of input:

Input image (**I**), method =1, k = 3, sigma = 2.0 and gamma = 0.1

Input	clsq_monkey.jpg_0.jpg 1 3 2.0 0.1

Input image (**I**), method =2, angle = 45 and steps = 20

Input	rl_theta=45_id=2.jpg 2 45 20

Example of output:

RMSE value in float format with 4 decimal places.

Output	29.9543

Submission

Submit your source code using the Run.Codes (only the .py file)

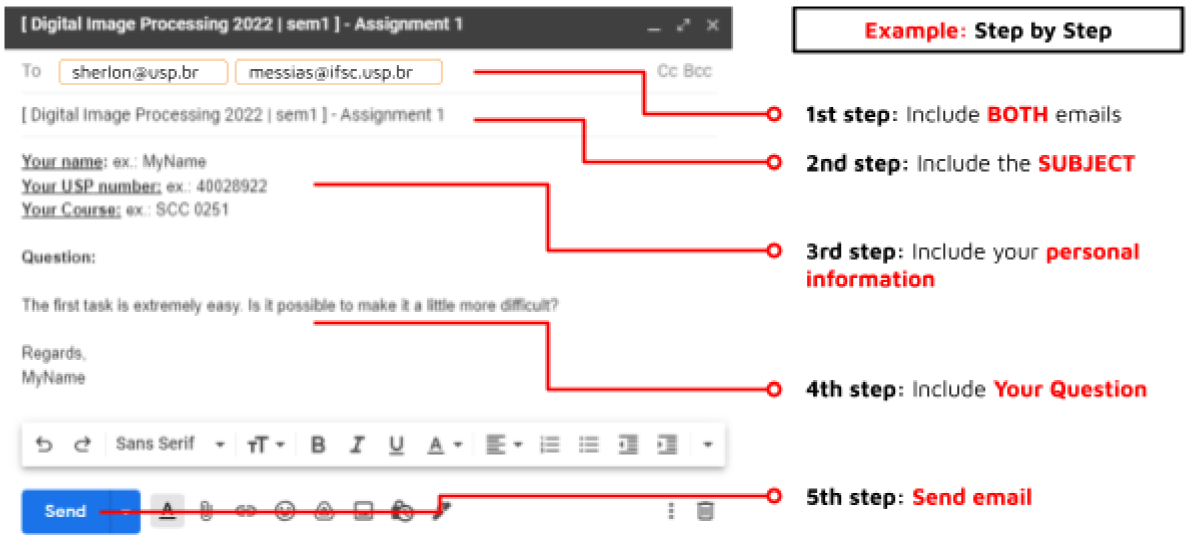
1. **Comment your code.** Use a header with name, USP number, course code, year/semester, and the title of the assignment. A penalty on the grading will be applied if your code is missing the header and comments.
2. **Organize your code in programming functions.** Use one function per method.

Contact

If you have any questions, contact us by sending an email following the five steps below:

1st step: Include **BOTH** emails, sherlon@usp.br **and** messias@ifsc.usp.br.
2nd step: Include the subject **exactly** like this:
Subject: "**[Digital Image Processing 2022 | sem1] - Assignment 4**"
Do not change the initial part (**black**).
Replace the final part with the topic you are interested in (**red**).
3rd step: Add your personal information to help us find your submissions in Run.Codes and E-Disciplinas quickly.
4th step: Formulate your question in detail. Include your implementation and/or screenshots if necessary.
5th step: Send email and wait. We will respond as soon as possible.

Example of Email:


[Digital Image Processing 2022 | sem1] - Assignment 1
To sherlon@usp.br messias@ifsc.usp.br
[Digital Image Processing 2022 | sem1] - Assignment 1
Your name: ex.: MyName
Your USP number: ex.: 40028922
Your Course: ex.: SCC 0251
Question:
The first task is extremely easy. Is it possible to make it a little more difficult?
Regards,
MyName

Example: Step by Step
1st step: Include **BOTH** emails
2nd step: Include the **SUBJECT**
3rd step: Include your **personal information**
4th step: Include **Your Question**
5th step: **Send email**