B31SE Image Processing: Assignment 1

Hugo Millet
27/01/2023
B31SE: Image Processing
MSc. Robotics
Electronical Electronic and Computer Engineering

---Task 1: Non-linear Image Filtering---

The main part of our code are the several intricated loops which are in order:

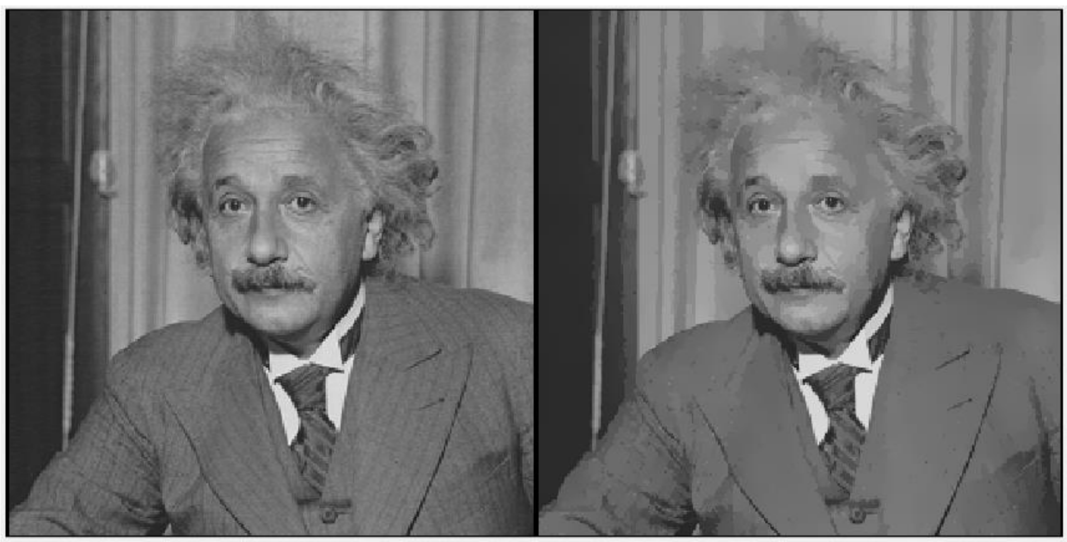
- The loop for the number of iterations
- The loop for browsing the pixels of the image
- The loop for calculating terms using the neighboring pixels of the current pixel (filter loop)

It is in this filter loop that we do our calculations, computing the following formulas:

$$I_{n+1}(x,y) = \sum_{i,j=-1}^{1} w_{ij} I_n(x+i, y+j) / \sum_{i,j=-1}^{1} w_{ij},$$
$$w_{ij} = \exp\{-k|I_n(x,y) - I_n(x+i, y+j)|\}$$

At the end, we update the current image used in the iterative process.

Here is the result of the computation for k = 150 and N = 50:



As expected, the filter removes the small details of the image, such as the stripes of the costume or the hair. Therefore, it increases the sharpness of the image, between the filter erases the small variations, making the big variations more visible.

Increasing N will produce a more intense result, as the number of iterations of the process will be higher.

On the contrary, increasing k will produce less intense results, as it is expressed as a negative power in the formula for the wij coefficient. Increasing k will thus decrease these coefficients, and therefore decrease the intensity of the filter.

---Task 2: Low-light image enhancement---

The process to enhance the light of the image is made of several calculations transposed in the code :

Formula	Code	Explanation

$T(x,y) = \max_{C\in\{R,G,B\}} L_C(x,y).$	`T2 = zeros(rows, cols);`
	`for i = 1:rows`
	`    for j = 1:cols`
	`        T2(i,j)= max(I2(i,j,1:3));`
	`    end`
	`end`
		We create a black and white image where each pixel is the maximum value of RGB for each pixel in the original color image

Let $U(x,y)$ be obtained from $T(x,y)$ by applying the image filtering	`U = nonlinearfilter(T2);`	We apply our filter from part 1 to smoothe the image

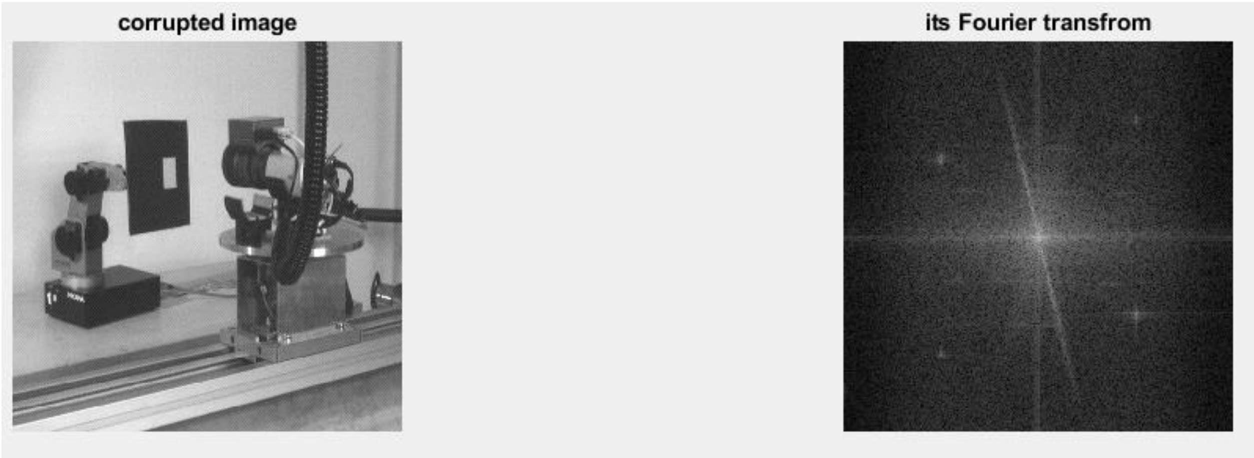
$E(x,y) = \frac{L(x,y)}{U(x,y)+\varepsilon},$	`epsilon = 0.1;`
	
	`E = I2;`
	`for i = 1:rows`
	`    for j = 1:cols`
	`        E(i,j,:) = E(i,j,:)/(U(i, j)+epsilon);`
	`    end`
	`end`
		Each RGB value from our original image is divided by the U image. As the U image is a smoothed version of a very dark image, its value is small, and therefore dividing by it increase each RGB component, enhancing the light and colors of the image

Here are the results on some dark images:



---Task 3: Image filtering in frequency domain---

We have a corrupted image and its Fourier transform :

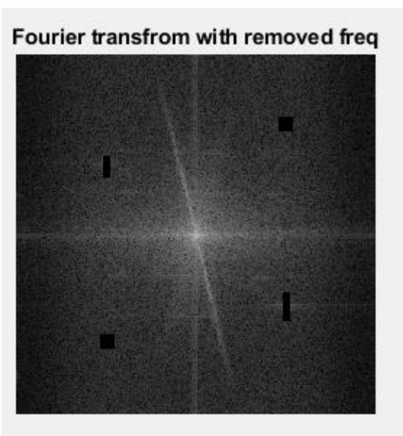


Our goal here is to remove the four little crosses that causes the corrupting stripes.

To do so, we located them, and applied rectangle filter with two intricated for loops, as shown in the example of one of the four removing loops :

```
for i = 125:135
    for j = 145:175
        imf(j,i) = 0;
    end
end
```

We check the result on the Fourier transform:



The main parts of each cross has been removed. We can convert back our image in its original form with the inverse Fourier Transform:

The stripes have been removed. As we have suppressed some useful frequencies in the process, the quality of the image has been a bit degraded, but the result is overall satisfying. To obtain better results, we could have applied more precise filters, in the shape of each cross for example.