Dpto. Ingeniería de Sistemas y Automática
Universidad de Málaga

COMPUTER VISION

EXERCISE 1.a: Introducing mVision

Concepts: Matlab GUIs

In this first exercise we are going to enhance and smooth images. Let's play a bit with mVision to gain insight into them, and then implement some code:

1. Launch the main mVision GUI by introducing *mVisionGUI* in the command window. Then launch the first GUI, *Smoothing*.
2. Load an image (for example, **kids**), and try the three available smoothing techniques: *Averaged environment*, *Gaussian filter* and *Median filter*.

Image 1

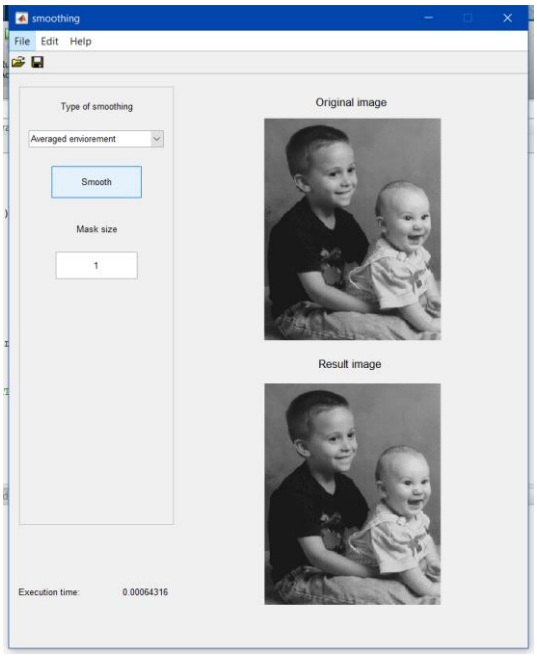

Image 2

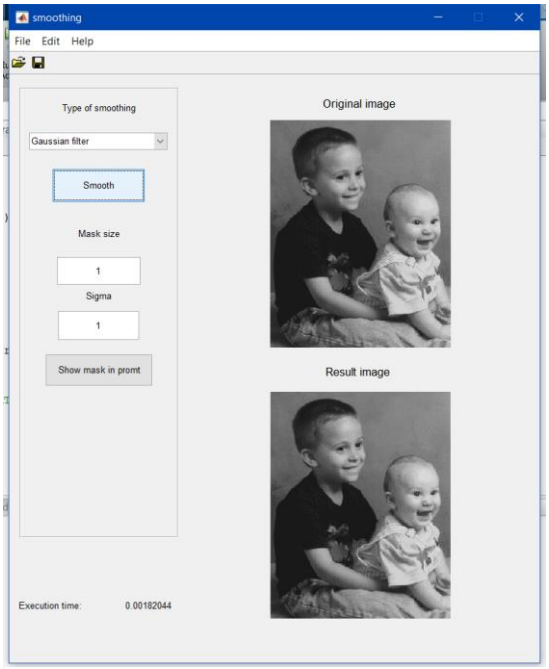

Image 3

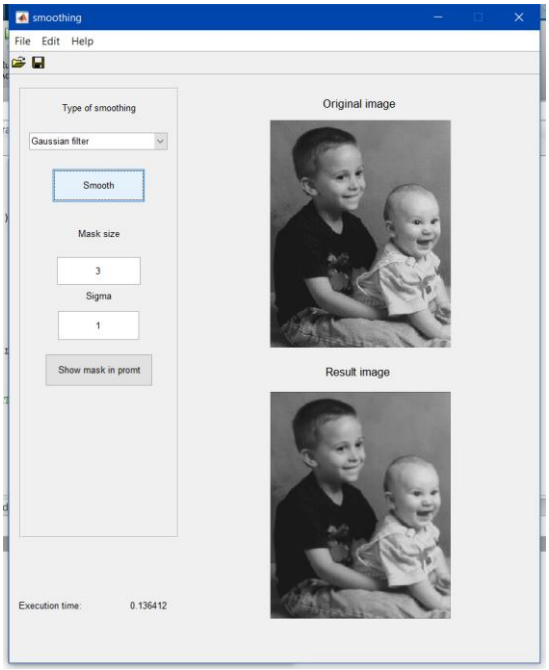

Image 4

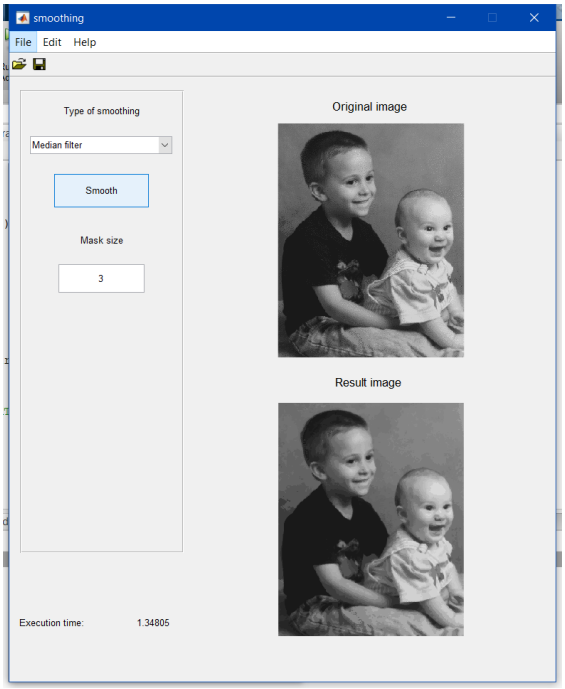

Image 5

a. What is the effect of changing the mask (kernel) size?
   If we increase the kernel size, the image will get more blurred in the edges, as we see in the odd numeration images which they are more blurred in the edges than the even numeration images.
b. Which technique is the slowest (largest execution time)? Why?
   The slowest is the median filter because the algorithm haves a high computational cost.

3. Now launch the second GUI, *Enhancement*.
4. Load the images **kids**, **lily** and **blood**. Which one show the highest quality?

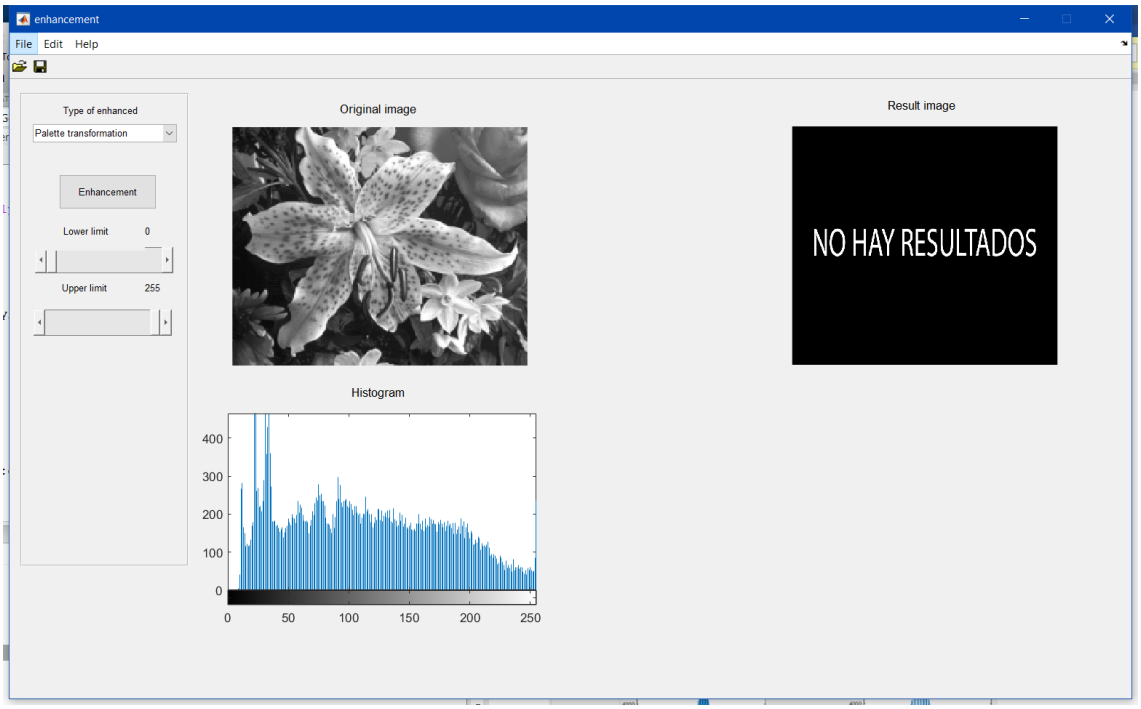

Image 6

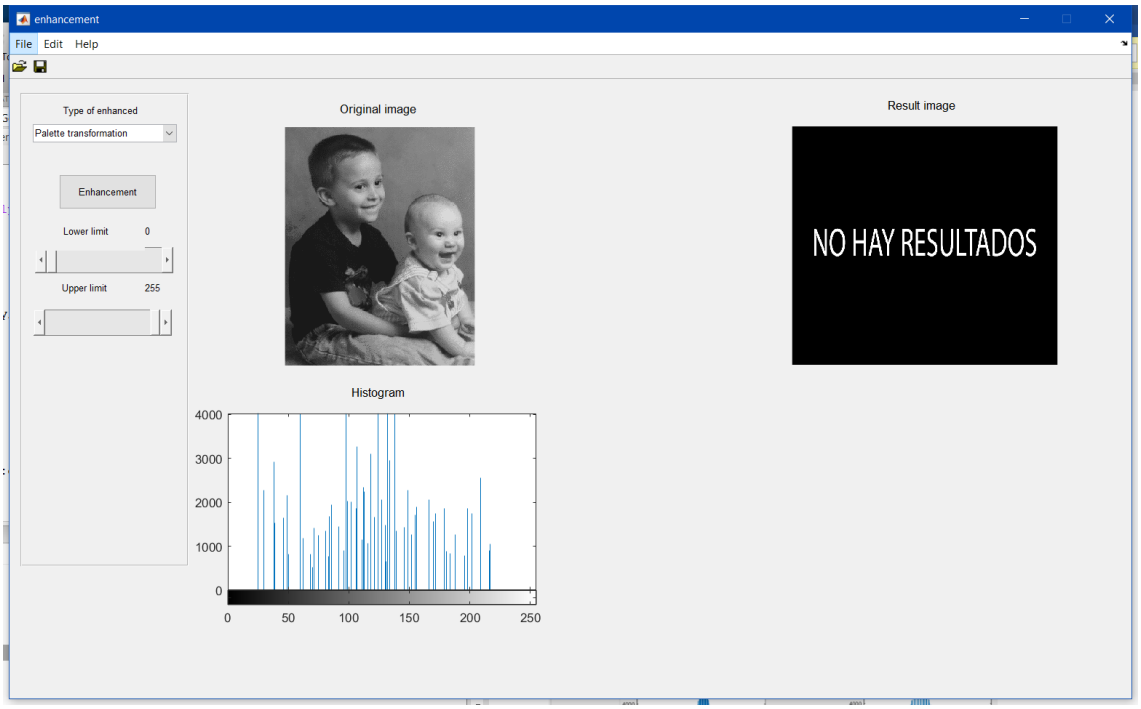

Image 7

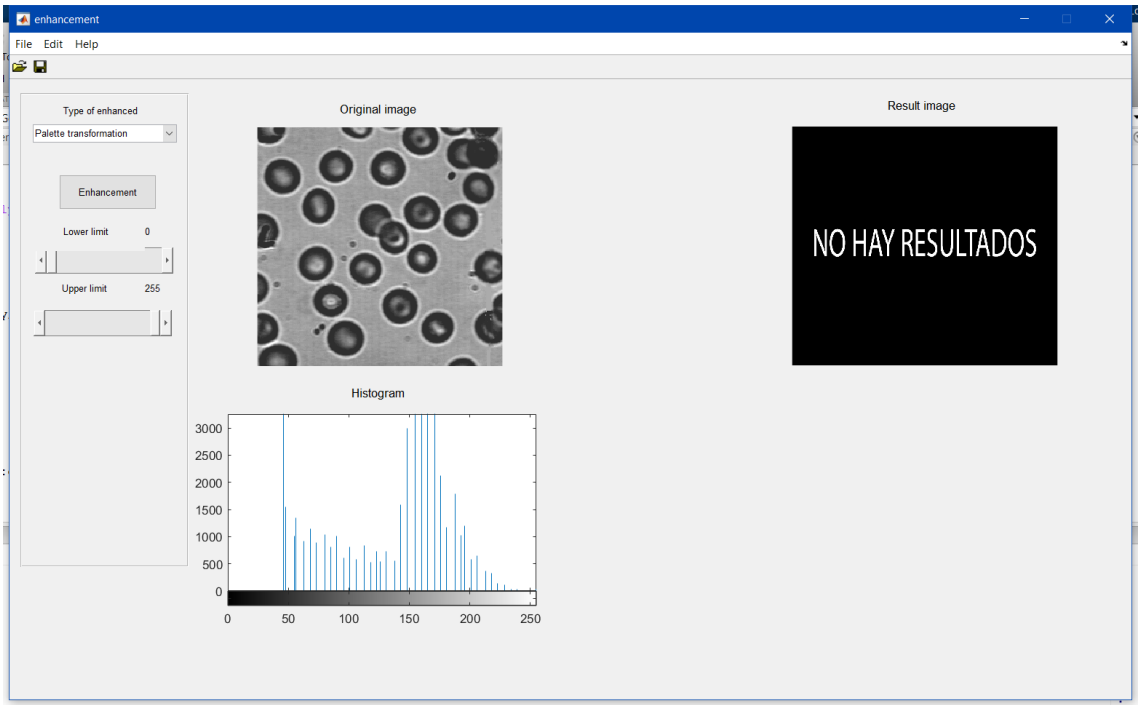

Image 8

The image with the most quality is the lily because in his histogram is more distributed than the other ones.

5. Finally, play a bit with the two implemented enhancement techniques: *Palette transformation* and *Histogram equalization*, and discuss an example of image enhancement that you consider interesting.

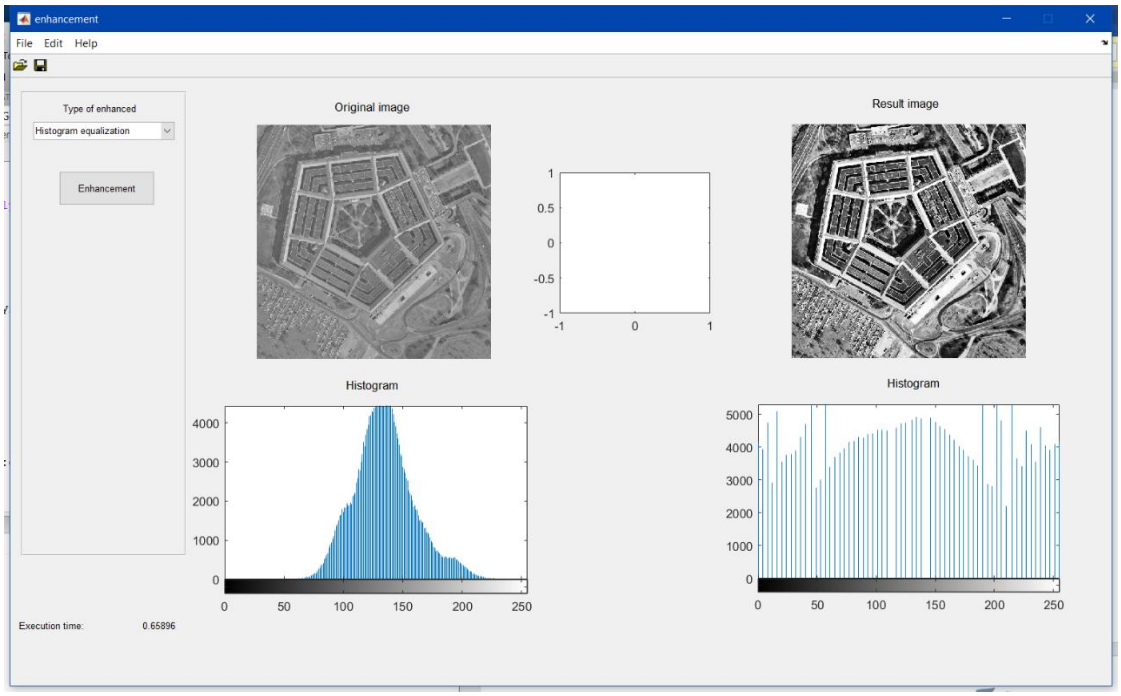

Image 9

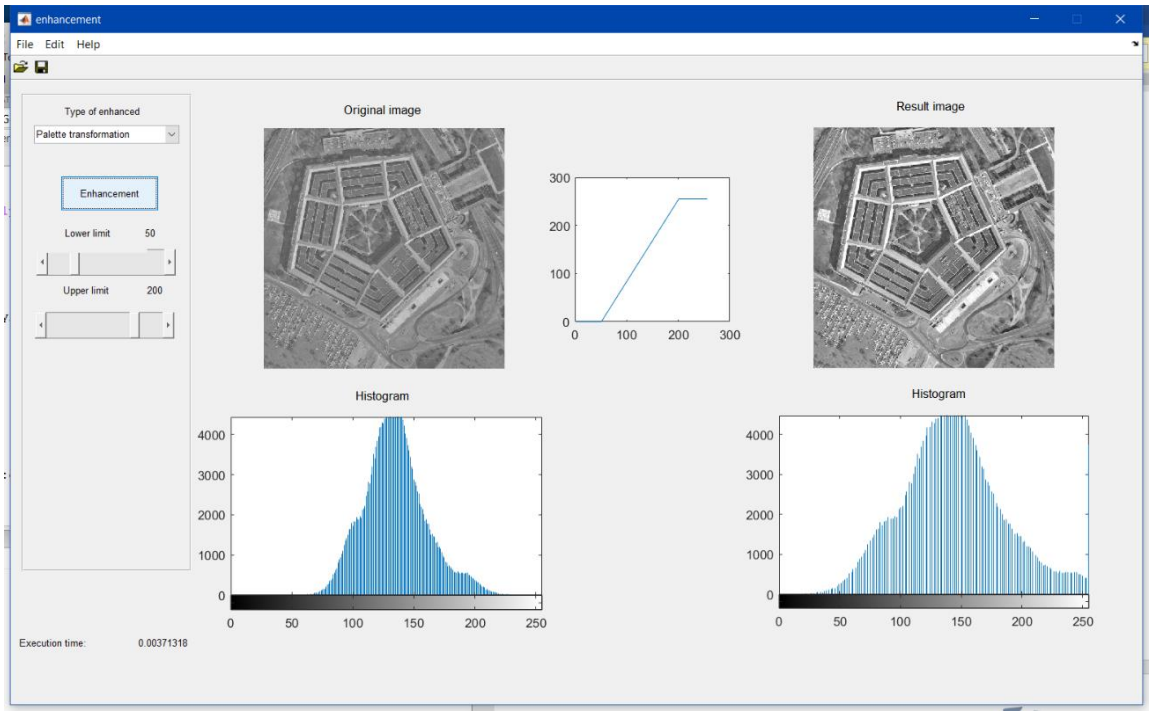

Image 10

As we see in both images, if we equalized the image, we have a more contrasted image than if we use a LUT. However, with a LUT we can modify the image as we want, even to make it negative.

Dpto. Ingeniería de Sistemas y Automática
Universidad de Málaga

EXERCISE 1.b: Image enhancement

Concepts: LUT, equalization and histogram specification

Implement a Matlab code showing a figure with 8 subplots (2x4), including the following:

1. Image **rice.tif** and its histogram.
   ```
   im = imread('test_images\rice.tif');
   imshow(im);
   imhist(im);
   ```

2. Image **rice.tif** enhanced with a LUT (to design analyzing the initial histogram) and the resultant histogram.
   ```
   im2=imadjust(im,[50/255,200/255],[0,1]);
   imshow(im);
   imhist(im);
   ```

3. Equalized image and its histogram.
   ```
   im3=histeq(im);
   imshow(im);
   imhist(im);
   ```

4. Image with a specified histogram and the resultant histogram. Hint: The specified histogram can be retrieved from other image.
   ```
   im_aux=imadjust(im,[0,1],[1,0]);
   im4=histeq(im,imhist(im_aux));
   imshow(im);
   imhist(im);
   ```

Results

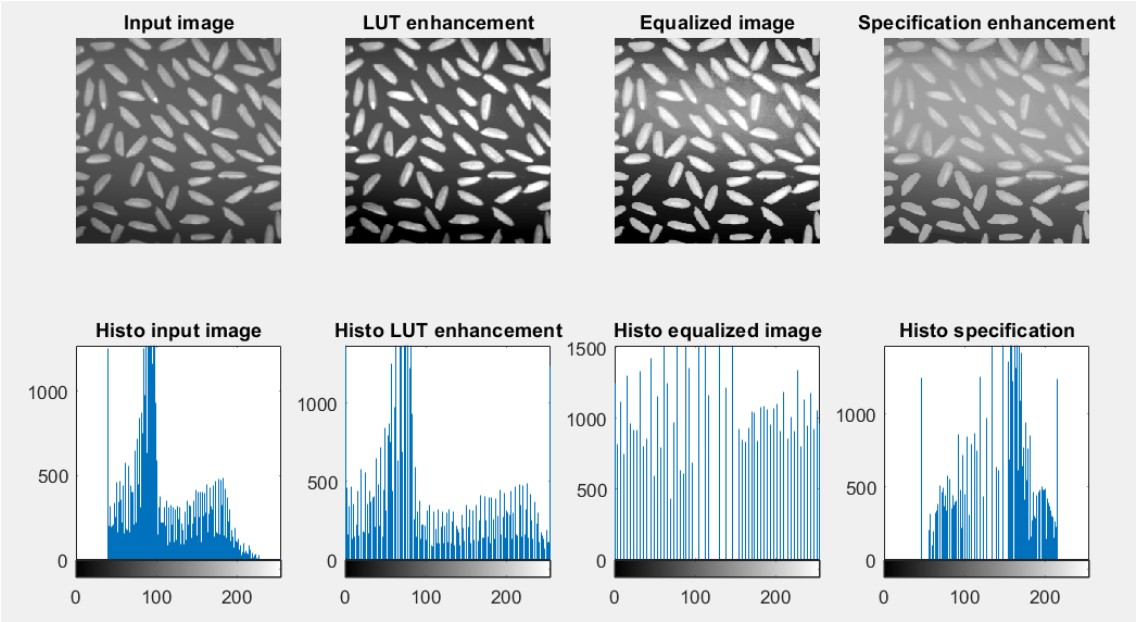

Image 1

EXERCISE 1.c: Image smoothing

Concepts: Noise in the image and filter types.

Implement a function that, given the name of an image, displays two figures (2x2) with the following:

1. Image with *Gaussian noise*, and its smoothed image employing a *Gaussian filter* with σ=0.5, a *Median filter*, and a *Neighborhood averaging filter* with a 3x3 kernel.
2. Repeat the previous images but with *salt & pepper* noise.

Analyze the obtained results.

Results (with **blood.tif**)

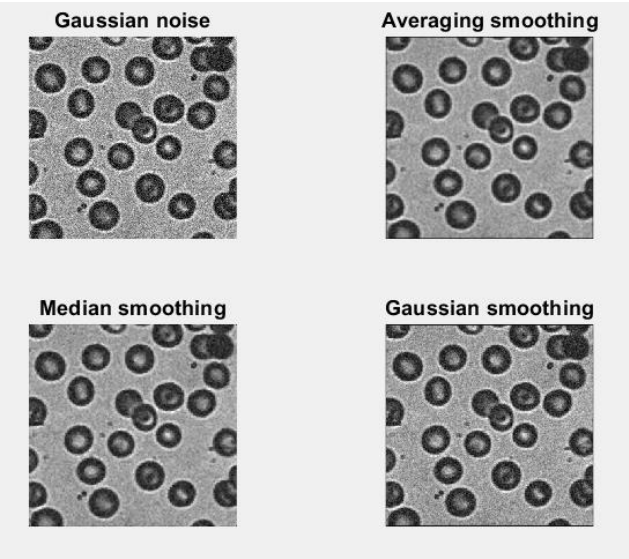

Image 1

We can see, with an image with gaussian noise, the best option is to do a gaussian smoothing. However, in this case, the averaging smoothing have a decent result.

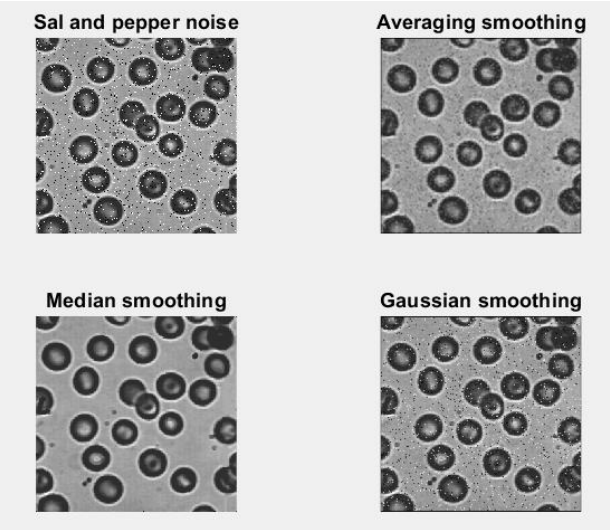

Image 2

We can see, with an image with salt and pepper noise, the best option without doubt is the median smoothing.