

# Noise Reduction Project on MATLAB

I implemented this project using two of the most famous filters Median filter and Order-Statistics filter

## Software used:

1. MATLAB R2016a
2. Sublime Text Editor

## Environment:

1. Windows 10 Operating System
2. 6 GB Ram with an integrated AMD graphics card and Corei5 processor

## Project Details:

### Files Attached:

- Noise Reduction (Folder) as a Zip File
  1. code.matlab
  2. images (Folder)
    - a. baboon.png (grayscale)
    - b. lena.png (grayscale)
    - c. forest.jpg (RGB image)
    - d. lion.jpg (RGB image)

### Warnings before running code:

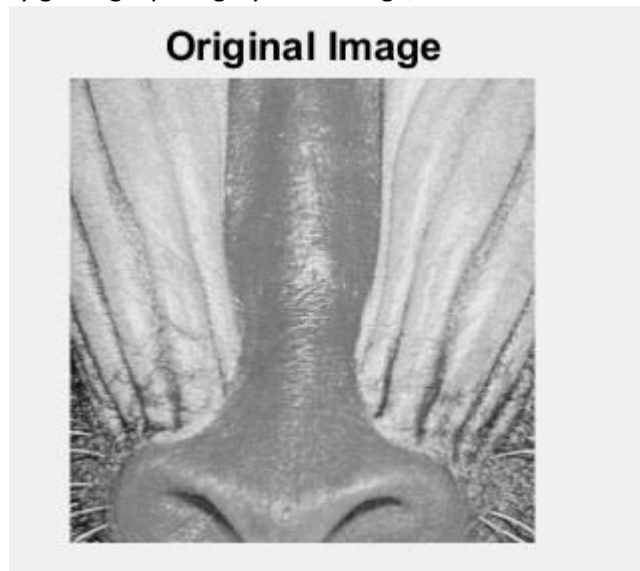
1. You must specify a directory to the folder attached (2. images) **Line 11**  
ex: `C:\Users\Moustafa\Documents\MATLAB\Noise Reduction\images`
2. You must specify a file name of the image you choose **Line 12**  
Ex: `baboon.png`
3. Put your own parameters as you wish (refer to “Details about filters used” section below)

### Details about filters used:

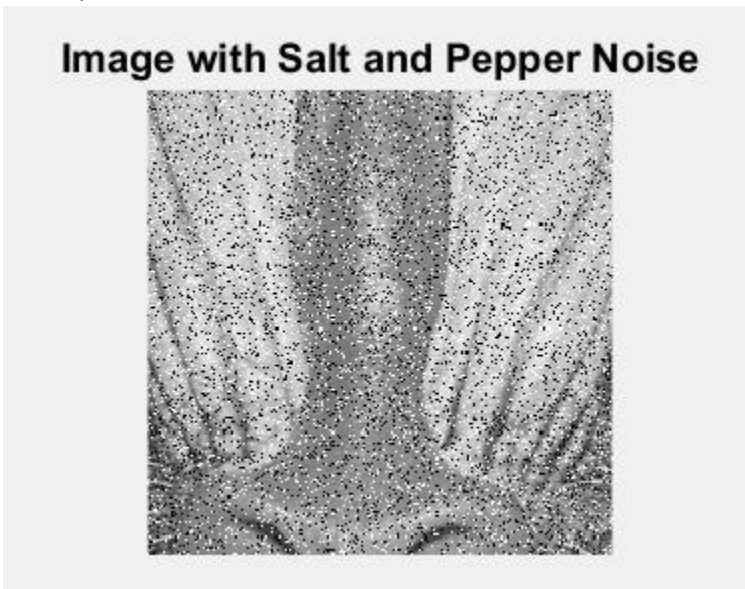
1. Median Filter
  - a. Function name: medfilt2
  - b. Parameters: **Lines 83, 84, 85, 132**
    - i. Image itself: a single channel image or just one channel out of an RGB image
    - ii. Filter size: an array of two elements that describes the two dimensions of the filter
  - c. Description: Found in documentation as referenced [\[1\]](#)

d. Experiment:

By getting a pure gray scale image,

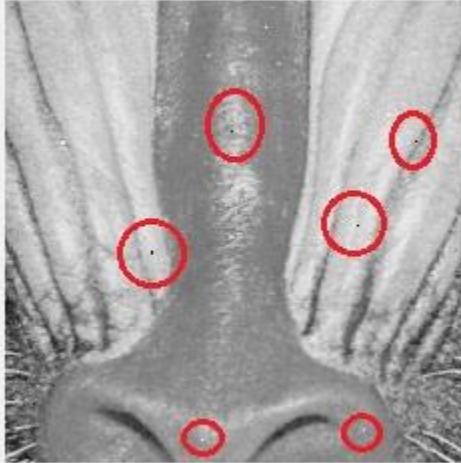


And then apply a salt & pepper noise using the MATLAB noise function (imnoise) with density of 15%



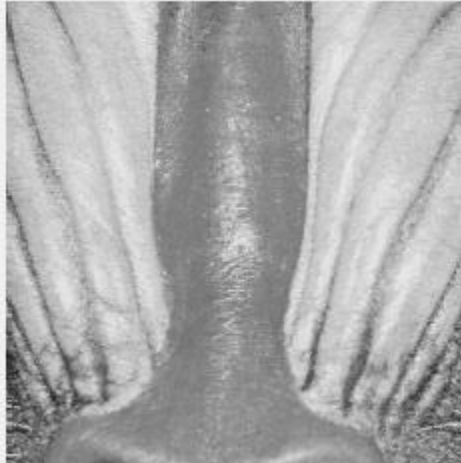
Then applying median filter on the noisy image using a filter size of **3x3**, we get a relatively good result of a restored image but if you can notice the red circles, there still some pepper and salt noise left without being restored.

### Restored Image using Median Filter



By changing the filter size to **5x5**, we get more pleasant results solving these left-overs

### Restored Image using Median Filter

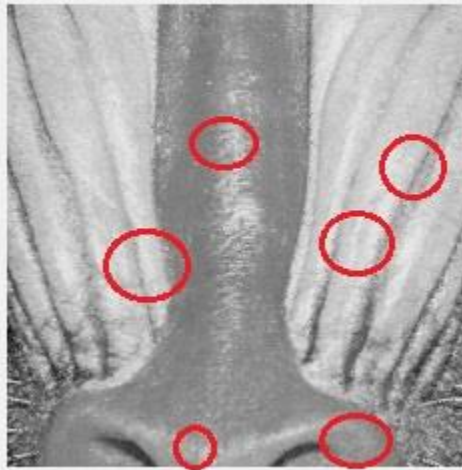


## 2. Order-Statistics Filter

- a. Function name: `ordfilt2`
- b. Parameters: **Lines 88, 89, 90, 134**
  - i. Image itself: a single channel image or just one channel out of an RGB image
  - ii. Filter order: an integer number that describes the nth order that we take out of a statistical series and that replaces the pixel
  - iii. Domain or Neighborhood
- c. Description: Found in documentation as referenced [\[1\]](#)
- d. Experiment:

By applying this filter with order of **10** on the same noisy grayscale image in the previous median filtered image (red circles)

### Restored Image using order-statistics Filter



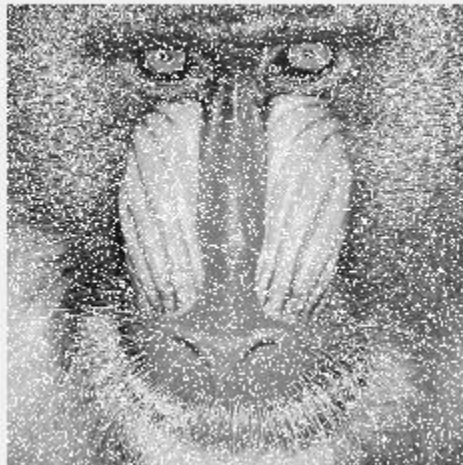
But when we change the order to 5, there appears some left-over pepper noise while salt noise is fixed

**Restored Image using order-statistics Filter**



And when we change the order to 50, there appears a lot of left-over salt noise, while pepper noise is fixed

**Restored Image using order-statistics Filter**



Which leaves us by choosing the intermediate value of **10**

I leave you then with some experiments those I've done and comparison between filters in individual images (Zoom-in to notice the difference between results)

MF-3x3

OS-10

Colored Image

Original Image



Image with Salt and Pepper Noise



Restored Image using Median Filter



Restored Image using order-statistics Filter



MF-3x3

OS-5

Colored Image

Original Image



Image with Salt and Pepper Noise



Restored Image using Median Filter



Restored Image using order-statistics Filter





MF-3x3

OS-25

Colored Image

Original Image



Image with Salt and Pepper Noise



Restored Image using Median Filter



Restored Image using order-statistics Filter



MF-5x5

OS-10

Colored Image

Original Image



Image with Salt and Pepper Noise



Restored Image using Median Filter



Restored Image using order-statistics Filter



## References:

1. [MATLAB Documentation](#) [1]
2. <https://www.mathworks.com/matlabcentral/answers/56515-noise-removal-in-image>
3. <https://www.mathworks.com/matlabcentral/answers/45268-noise-removal-from-colored-image>