

Lab 3

Averaging Filters

1. Filtering:
 - a. Read an image of your choice.
 - b. Apply salt & pepper noise with density =0.05
 - c. Apply median filtering algorithm using a mask of size 3*3
 - d. Compare your results with python function ***Skimage.filters.median***
 - e. Apply **Gaussian Filtering** to the noisy image image with the following conditions:
 - i. Lengths [3 3] [7 7] for sigma = 0.5.
 - ii. Sigma(s) 8 and 0.2 for length = [3 3].
 - iii. After understanding the effect of length and sigma. Choose the best combination that fits your image. Needs Justification.

edgex := (window width / 2) rounded down
 Usefull New Functions and Attributes
 edgey := (window height / 2) rounded down
 for x from edgex to image width – edgex

Name	Attribute or Function	Usage
<pre> for y from edgey to image height – edgey { round allocate colorArray[window width][window height] for fx from 0 to window width for fy from 0 to window height colorArray[fx][fy] := inputPixelValue[x + fx - edgex][y + fy - edgey] sort all entries in colorArray[[]] outputPixelValue[x][y] := colorArray[window width/2][window height/2] } </pre>	Function	
np.zeros	Function	To generate matrix with a given shape all elements with zero values
range	Function	To generate range of indexes from low range to high range with an optional step
gaussian	Function	To apply gaussian filter on an image.
np.median	Function	To get median value. <i>To get the median value for all the matrix (not for a certain axis), provide parameter axis=None</i>
Skimage.filters.median	Function	To apply median filter on an image.

To make the range of a gray-scale image uint8 (from 0 – 255), use `astype(np.uint8)`.