

CS 4650/7650 ECE 4655/7655 Digital Image Processing

Homework 3 – Noise Filtering [100 pts + 10 bonus pts]

Out: Thursday Oct 2

Due: Tuesday Oct 14

The goals of this assignment are:

- Experimenting with various image noise filtering approaches
- Quantitative analysis of the results
- Efficient implementation of adaptive median filter described in class

Programming language: Python

Assignment:

Part 1 [10 pts] Mean Square Error (MSE) implementation.

MSE calculates the average squared difference between the corresponding pixels of two images. MSE provides a numerical value indicating similarity of two images and it is widely used to assess image quality. MSE between two images is defined as:

$$MSE = \frac{1}{MN} \sum_{x=1}^M \sum_{y=1}^N (I1(x,y) - I2(x,y))^2$$

err=MSE(I1,I2)

Part 2 [20 pts] Image noise filtering using built-in functions and discussion of the results.

Use built-in OpenCV and/or SciPy functions to apply 2D Gaussian filter and 2D median filter to the given test images. Generate the following table using MSE function written above.

| | MSE | | | | |
|-------------|-----------------------|----------------------------|----------------------------|---------------------|-----------------------|
| | Noisy (not processed) | Gaussian Filter sigma=2 | Gaussian Filter sigma=7 | Median Filter (7x7) | Median Filter (19x19) |
| Test1Noise1 | | | | | |
| Test1Noise2 | | | | | |

Note: All MSE values will be computed with respect to the original clean (not noisy) image.

Report should include:

1. Original image
2. Noisy images
3. The 8 filtered images generated above

4. MSE table shown above
5. Interpretation of the results (i.e explain if the approach generated acceptable results, why or why not?)

Part 3 [70 pts] Implement Adaptive Median Filter as described in class and discussion of the results. See lecture notes Lec7_Filtering_Adaptive_Integral_Histogram.pdf on CANVAS.

Note 1: implementation should be efficient. Consider using built-in 2D median filter.

Note 2: program speed-up using Numba Jit or Njit is not allowed.

`[I_filtered]=AdaptiveMedian(I_noisy,Smax)`

| | MSE | | | Processing Time | |
|-------------|-------|---|---|---------------------------------|-----------------------------------|
| | Noisy | Adaptive Median Smax = [7x7] (3x3 to 7x7) | Adaptive Median Smax = [19x19] (3x3 to 19x19) | Adaptive Median Smax = [7x7] | Adaptive Median Smax = [19x19] |
| Test1Noise2 | | | | | |
| Test2Noise2 | | | | | |

Report should include:

1. Original image
2. Noisy images
3. The 4 filtered images generated above
4. MSE and processing time table shown above
5. Interpretation of the results (i.e explain if the approach generated acceptable results, why or why not?)

Bonus [10 pts]: Plot number of finalized (reconstructed pixels at each filter size)

Submission Instructions: Submit electronic version from Canvas (<https://courses.missouri.edu/>). Submit the following directories and files:

1. src - should contain all program files
2. A report in PDF format (not Word or other formats that are difficult to handle with images and equations). Note that you can print to PDF.
3. Images (if any) should be in a separate folder - for example if you tried your own images