CMPG-767 Image processing and Analysis
Project 2

Student: Emeshe Sotak SA1/1

The purpose of this project is to compare the efficiency of spatial domain linear filters

1. Design the following Matlab functions:

a) A function evaluating a root mean square error (RMSE, standard deviation) and peak signal-to-noise ratio (PSNR) between two images.
b) A function for adding Gaussian additive noise to an image. This function shall accept as parameters a clean image and a coefficient determining a noise standard deviation as a fraction of an image standard deviation.
c) A function performing spatial domain linear filtering of an image with a 3x3 local neighborhood window accepting a filter kernel (3x3 matrix) as a parameter. Use the mirrorImage.m function, which was shared with you, for taking care of boundary effect or design your own function for taking care of it (**t**he latter gives you 30 extra credit points).

2. Design a Matlab script, which utilizes the following (use functions, which you designed here and in Project 1):

a) Measures mean and standard deviation of an image;
File: image_statistical_analysis.m (from project 1)

b) Adds Gaussian noise to an image;
File: additive_gaussian_noise.m

c) Applies linear filtering with a given kernel to a noisy image;
File: filtering.m

d) Measures standard deviation and PSNR between the noisy image and the clean image and between the filtered image and the clean image.
File: rsme.m, psnr.m

3. Choose an image *f* (*x*, *y*)

```
Min - 0
Max - 198
Mean - 8.195727e+01
Standart deviation - 2.681390e+01
Variance - 7.189852e+02
Signal-to-noise ratio - 3.056522e+00
```

4. Generate Gaussian noise η(*x*, *y*) with the standard deviations 0.2σ and 0.3σ where σ is the standard deviation of the initial image.

5. Create two noisy images by adding the noisy fields to the image according to:

$$g(x, y) = f(x, y) + \eta(x, y) - m$$

0.2σ

0.3σ

6. Filter your noisy images using a linear filter determined by the following kernel and evaluate RMSE and PSNR for your filtered image.

$$\frac{1}{16}\begin{pmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{pmatrix}$$

0.2σ

0.3σ

7. Following the idea of smart filtering (Slides 6 and 7 of Lecture 5) try to find another kernel outperforming the one from Task 6 in terms of PSNR.

8. Save all your images and create a table with the summary of your results (Kernel 1 (which was given), Kernel 2 (which you found), and corresponding RMSE & PSNR)

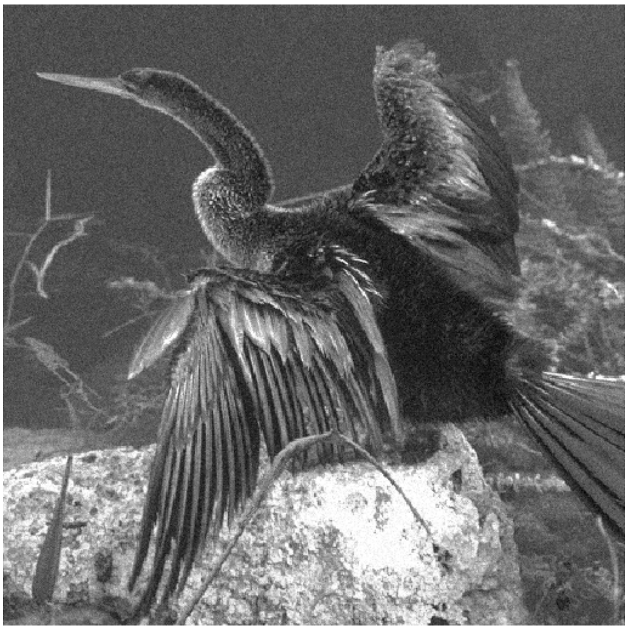
	Original and noised image		Original and filtered image	
Standard deviations	**0.2σ**	**0.3σ**	**0.2σ**	**0.3σ**
PSNR	**3.35**5443e+01	**3.00**3305e+01	**3.32**6336e+01	**3.25**8603e+01
RMSE	**5.35**5730e+00	**8.03**3187e+00	**5.53**8243e+00	**5.98**7404e+00

9. Repeat steps 3-8 for another image.

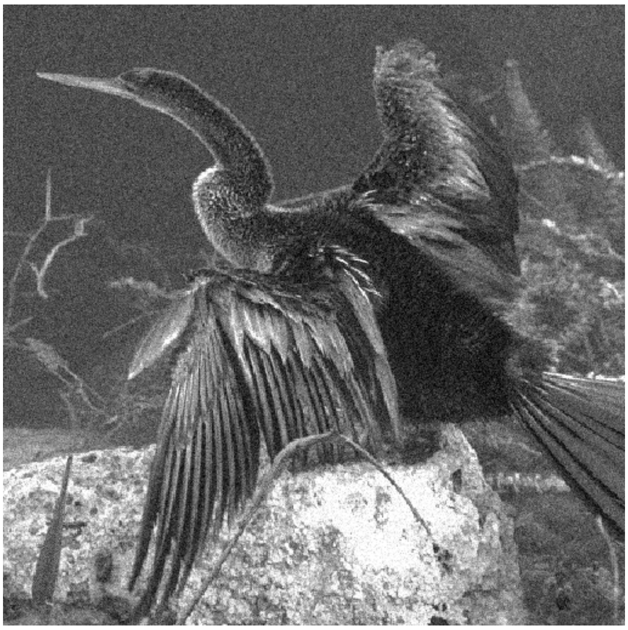
```
Min - 25
Max - 255
Mean - 1.035874e+02
Standart deviation - 4.981850e+01
Variance - 2.481883e+03
Signal-to-noise ratio - 2.079297e+00
```

Gaussian noise

0.2σ

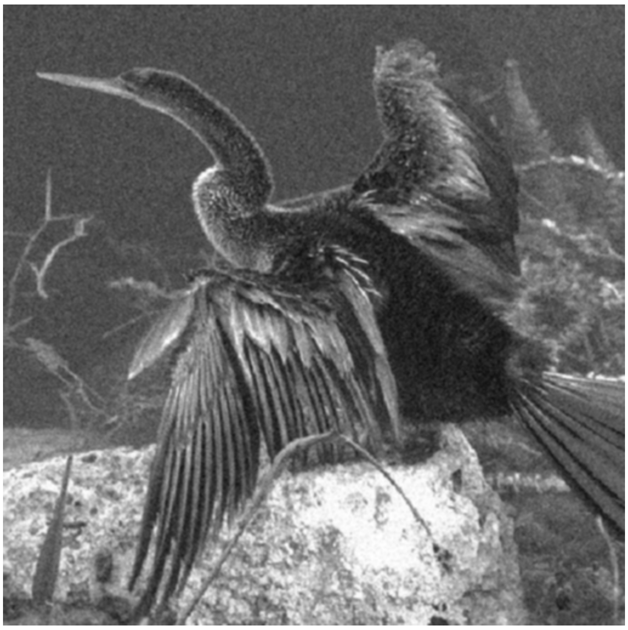


0.3σ



Linear filter

0.2σ **0.3σ**



	Original and noised image		Original and filtered image	
Standard deviations	**0.2σ**	**0.3σ**	**0.2σ**	**0.3σ**
PSNR	**2.82**0968e+01	**2.47**1438e+01	**2.88**3089e+01	**2.79**9833e+01
RMSE	**9.90**9600e+00	**1.48**1907e+01	**9.22**5625e+00	**1.01**5368e+01