EE5175: Image Signal Processing - Lab 12 report

Ruban Vishnu Pandian V (EE19B138)

April 21, 2023

Contents

1 Non-local means (NLM) filtering

Aim: To de-noise a given noisy image using **Non-local means (NLM)** filtering algorithm and compare it with the output of a standard Gaussian filter. Also, the PSNR values for different parameters must be computed and analyzed. The images are given below:

Clean image

Noisy image

2 NLM filtering algorithm:

The central idea behind noise filtering is the fact that most of the noises are zero mean random variables. Hence, averaging over several captures of the same scene would give a cleaner image. However, when only one noisy image is available, this method is not possible. Instead, a local averaging is typically done to reduce the noise.

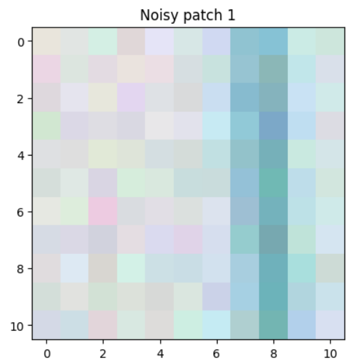
However in the process, the image also becomes smooth since averaging mixes intensities and hence, blur is introduced in the image. NLM filtering is an algorithm which effectively removes the noise while maintaining the sharpness of the image. The key idea is to average over neighbourhoods that are similar rather than a naive local averaging. The algorithm is defined as follows:

1. For every pixel **p** in the image, define a search neighbourhood characterized by the search radius W. The search neighbourhood would then be a $(2W+1)$X$(2W+1)$X3 RGB array with the central pixel being **p**.
2. Similarly, a similarity neighbourhood of radius W_{sim} is also defined for the pixel.
3. For every pixel **q** in the search neighbourhood, its similarity neighbourhood is obtained.
4. Let the unrolled similarity neighbourhood vectors be V_p and V_q. Then the weight for pixel **q** is given as: $W_p(q) = exp(\frac{-(V_p-V_q)^T(V_p-V_q)}{\sigma^2_{NLM}})$.
5. Once the weights for all the pixels are obtained, they are sum-normalized and used to compute the output intensity, i.e., a weighted average of the intensities of all the pixels in the search neighbourhood is computed. This is the output intensity of pixel **p**.

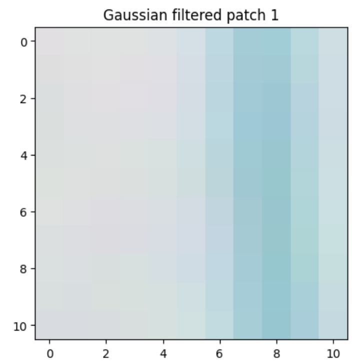
In this way, not all neighbouring pixels have the same contribution to the output. Pixels which are similar have more contribution. Hence, blur is reduced but noise reduction is still present.

3 Results on given images:

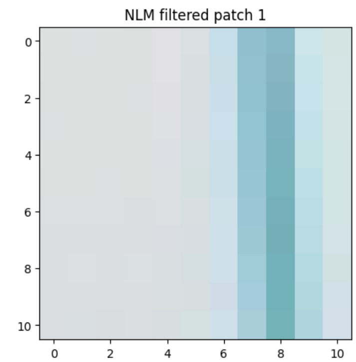
The NLM algorithm was used on the noisy image and the outputs are given below:



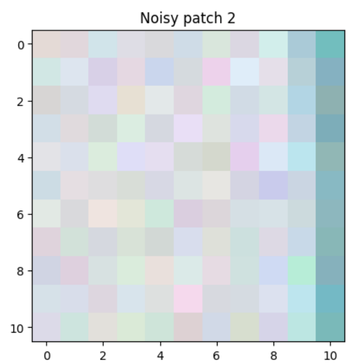
Noisy patch



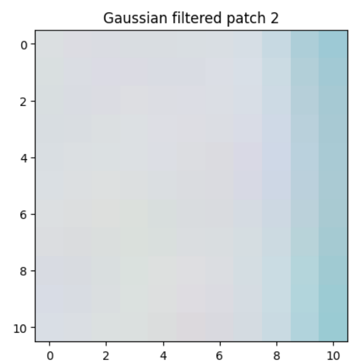
Gaussian filtered



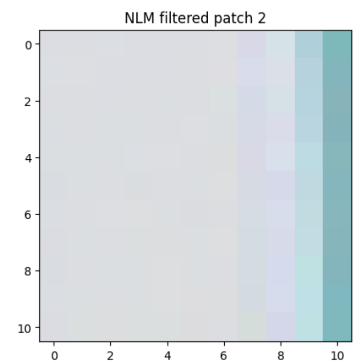
NLM filtered



Noisy patch

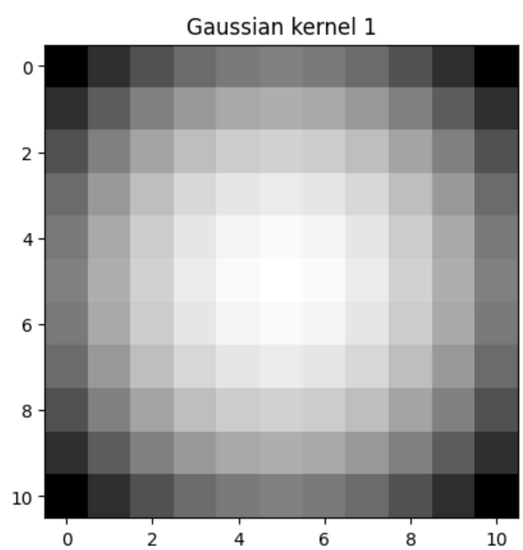


Gaussian filtered

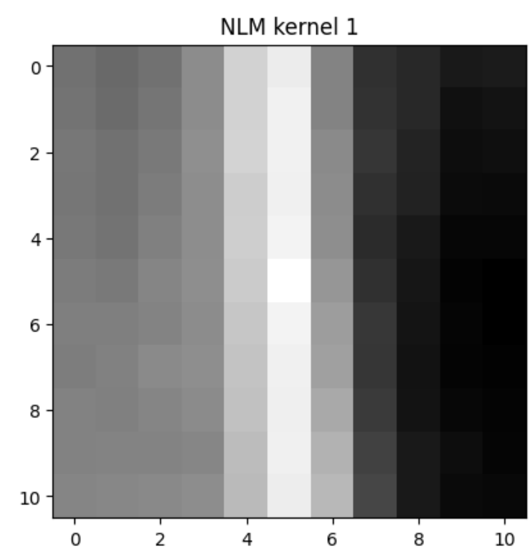


NLM filtered

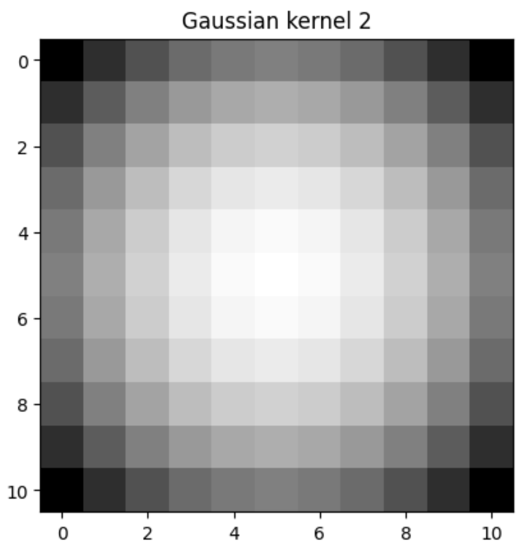
Clearly, the NLM filtered output looks more sharper than the Gaussian filtered outputs. The kernel/weight images are also given below on a logarithmic scale:



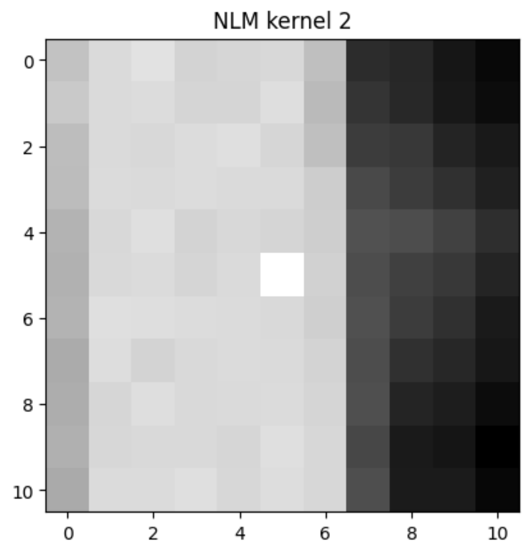
Gaussian kernel for location 1



NLM kernel for location 1



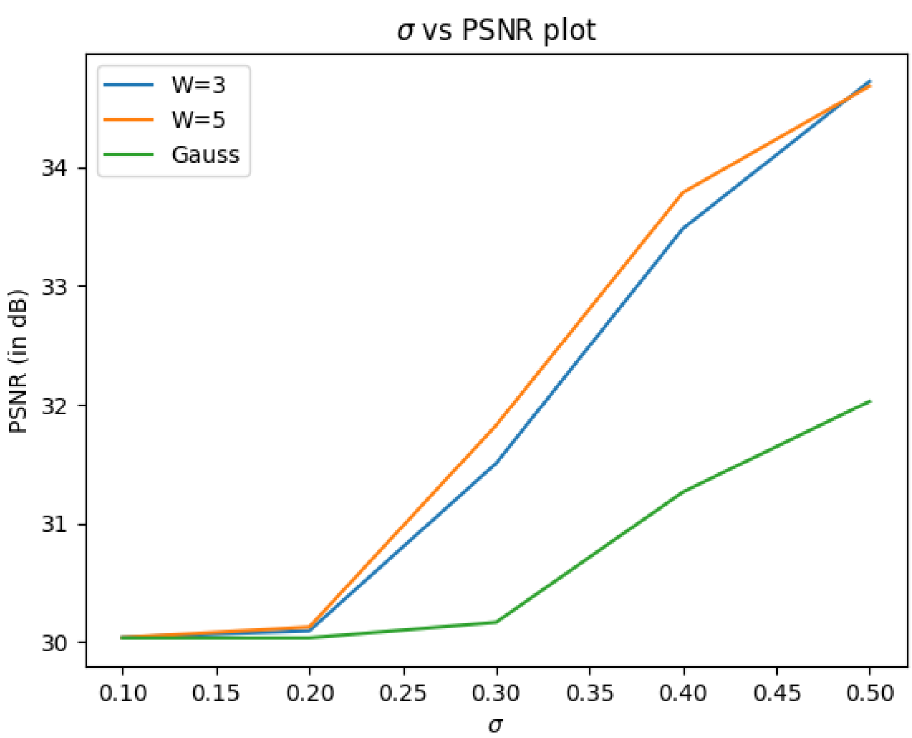
Gaussian kernel for location 2



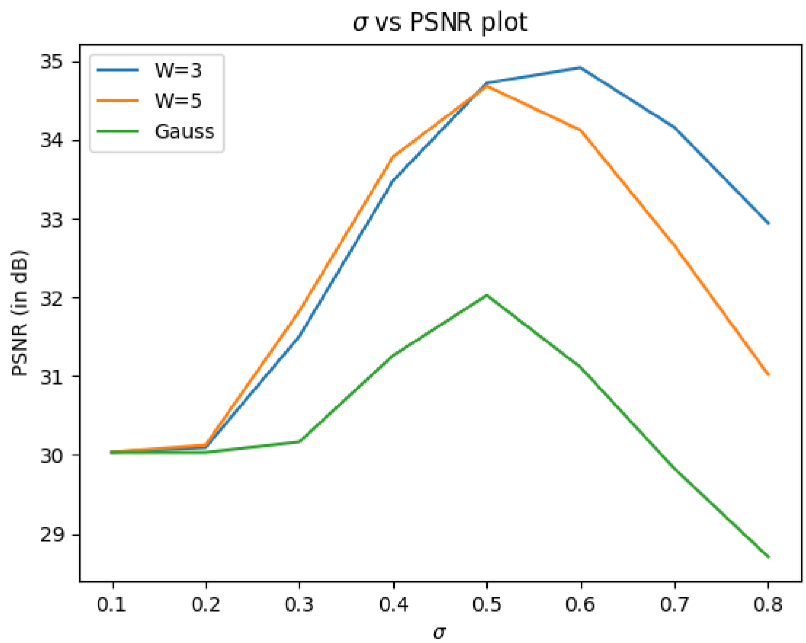
NLM kernel for location 2

The Gaussian kernel is the same for both locations since it is space invariant. The NLM kernels look very different signifying how the degree of similarity is very different at these locations.

The baseline PSNR for this test case is **30.033 dB**. The PSNR was computed for different σ_{NLM} values and the plot is given below:



As expected, the PSNR obtained by using the NLM filtering algorithm is better than that of the Gaussian filtering algorithm. When a bigger search window is used, the PSNR is better but after a particular σ_{NLM} value, it falls behind.



It is clear that beyond $\sigma_{NLM} = 0.5$, the PSNR starts to fall and $W = 3$ provides better PSNR than $W = 5$.

4 Observations and Conclusions

1. From the images of the patches and the kernels, it is evident that NLM kernels model the similarity present in the neighbourhood and hence, provide effective de-noising with very insignificant blur.

2. The PSNR obtained depends on the σ_{NLM} and W used. For σ_{NLM} from 0.1 to 0.5, the PSNR is better for $W = 5$ but it is the other way round for σ_{NLM} beyond 0.5.

3. Beyond $\sigma_{NLM} = 0.5$, PSNR starts to reduce. This can be explained from the fact that the effect of similarity starts to reduce in the expression $W_p(q) = exp(\frac{-(V_p - V_q)^T (V_p - V_q)}{\sigma_{NLM}^2})$, i.e., all search neighbourhood pixels are weighed almost the same.