CONCORDIA UNIVERSITY

GINA CODY SCHOOL OF ENGINEERING AND COMPUTER SCIENCE

COMP 6771

IMAGE PROCESSING

FINAL PROJECT REPORT

Submitted to: Dr. Yiming Xiao

By,

Anushka Sharma (40159259)
Sejal Chopra (40164708)

TABLE OF CONTENTS

PAPER REVIEW

Due to recent explosion in the number of digital images taken every day increasing the demand for more accurate and visually pleasing images that are inevitably degraded by noise creating a demand for image denoising techniques. In this paper[1], the authors have done a comparative study between four classic denoising algorithms (Gauss filtering, anisotropic filtering, Total variation, and Neighbourhood filtering) and non-local means. Whereas the second paper[2] proposes a type of non-local means technique called adaptive non local-means (ANL-means) algorithm which uses robust block classification in noisy images.

In Paper 1 an image is taken and all different approaches of denoising are applied one by one - Gauss filtering, anisotropic filtering, Total variation, and Neighbourhood filtering with decided standard deviation and respective parameters. A track is kept of the impact of similarities as a function distance which is used to control the weight of the patch which is used to calculate the NL mean value. Then method noise is calculated to compare the performance. The quality of the restored images is evaluated with noise of std 35 and 20, respectively. Finally, the mean square error of the restored images and the original images is also calculated to check the difference between them.

In Paper 2 an image is taken having zero mean AWGN having SD as n = 20, 30 and 40. Each pixel is taken, and blocks are classified by using SVD to the gradient field of each block. And on the type of region, it is computed if there is dominant direction and size of computed singular values. This approach is applied adaptively using K means clustering technique. Based on classification, matching window size is decided. On the same, rotated matching process is applied allowing more candidates of similar image blocks taking orientation angle into consideration. Laplacian operator is used to measure the correct value of noise level used in noise variance. Performance - the averaged PSNR results of the three denoised images are reported

It was observed in paper 1 that Nonlocal means algorithm with standard deviation of 20 works most efficiently keeping the texture and edges intact of the image which is reduced as the standard deviation is increased. Hence the mean square error of NLM algo is the minimum among all, proving the previous statement. Whereas in paper 2, when traditional Nonlocal and Adaptive nonlocal means algorithms are compared it is observed that later one is more efficient. For a set of seven representative test images corrupted by the zero-mean AWGN, three Gaussian noise patterns are generated and the PSNR results are reported. The average PSNR of the ANL-means scheme is approximately 2.98, 2.55, 3.70, 1.88 and 2.06 dB better than MF, GF, PDE, TV and NL-means, respectively. The ANLmeans achieves an average PSNR improvement of 0.63, 2.16 and 3.39 dB over the NL-means for V n = 20, 30 and 40, respectively. The experiments were relevant to show the efficiency of the proposed method for denoising. However, there is no comparison of complexity or computational cost of each of the methods. Also, the established values of the parameters of the compared models are not mentioned in the article. Looking at figure 3, it is not clear that it is part of the baboon image. Apart from that the result of the noise method for the baboon image is not shown, it will be interesting to observe if the estimated noise in that image contains some of the texture information. It should include future work and the challenges of the method. In paper 2, the approach of adaptive nonlocal means is very useful and gives substantial results than non-local means. However, the results could be more accurate if the sample size of images is increased. Along with that, more elaboration on the k-means approach used for block classification would be appreciated. The computational time of the ANL is higher than that of the NL-means by a factor of 3 or 4 due to larger matching windows. The block rotation operation can be effectively removed by the angle selection scheme reducing complexity. This is also mentioned in the paper itself.

Research paper link

1. Nonlocal means denoise
2. Improved_image_denoising_with_adaptive_nonlocal_means_ANL-means_algorithm

ABSTRACT

Images have been a crucial part of our lives for a long time now but capturing them have been inevitably degraded by different types of noises. There are many denoising techniques that have been proposed like Gauss filtering, anisotropic filtering, Total variation, and Neighbourhood filtering. Among all these denoising techniques Nonlocal means algorithm gives best results since it takes a mean of all pixels in the image, weighted by how similar these pixels are to the target pixel. The NL-means not only compares the grey level in a single point but the geometrical configuration in a whole neighbourhood. The quality of the denoised images is judged by the concept of method noise. It judges the difference between the denoised image and original image, hence should be minimum and having as little structure as possible for best results. The images are computed and analysed for all these denoising algorithms. The observations are made and mean square error is calculated. Mean square error also indicates that the estimate is closer to the original image. But these evaluation parameters only indicate the efficiency of the algorithms and cannot guarantee the complete removal of noises. The human eye is the only one able to decide if the quality of the image has been improved by the denoising method.

INTRODUCTION

Image denoising, whose main purpose is to estimate the original image by suppressing noise from a noise-contaminated version of the image, is one of the fundamental problems in the field of image processing and computer vision. Different intrinsic (sensor-related) and extrinsic (environment-related) circumstances, which are frequently impossible to prevent in real-world applications, can result in image noise. Due to the need of recovering the original image content for effective performance, image denoising is used in a variety of applications including image restoration, visual tracking, image registration, image segmentation, and image classification.

Although various algorithms have been put forth for the goal of image denoising, the issue of image noise suppression still remains unsolved, particularly when the photos are taken in unfavourable conditions with a high degree of noise. The reference paper lists several denoising methods, including Gaussian filtering, anisotropic filtering, total variation minimization, and neighbourhood filtering. The paper suggests a novel nonlocal means method for denoising.

Gaussian filtering blurs the images and features by using a mathematical function known as the Gaussian function. Based on the Gaussian distribution, the Gaussian Smoothing Operator computes a weighted average of the surrounding pixels. Whereas the Laplacian cannot be small, the gaussian method noise is negligible in harmonic regions of the image and very large near edges or texture. In flat areas of the image, the Gaussian convolution is therefore best, but edges and texture are obscured.

Anisotropic filtering is a method of filtering that alters the quantity of texture samples produced based on the angle that the surface to be displayed is at with respect to the camera. Surfaces or patterns that are inclined and further away from the camera appear better and crisper when anisotropic filtering is used as opposed to when it is not.

Total variation denoising is a method for eliminating noise (filter). It is founded on the idea that signals with excessive and potentially erroneous detail have high total variation, or that the absolute picture gradient integral has a large value. According to this theory, lowering the signal's overall variation—provided it closely matches the original signal—removes unnecessary complexity while keeping crucial characteristics like edges.

Neighborhood filters are frequently used to improve the images so that segmentation algorithms can work more effectively. By averaging the values of nearby pixels with a comparable grey level value, it recovers a pixel.
The issue with these filters is that when the values are noisy, comparing just the grey level values in a single pixel is not as reliable. Nonlocal means algorithm stands out in this context. Nonlocal filtering averages all of the image's pixels and weights them according to how similar they are to the target pixel. The NL-means compares the geometric configuration throughout the entire neighbourhood as well as the grey level at a particular place. Compared to local mean and other methods, this yields substantially more post-filtering clarity and less loss of detail in the image. For noise removal, the NL-means takes advantage of the spatial correlation over the entire image.

Each pixel's value is adjusted using a weighted average of pixels in its neighbourhood with comparable geometrical configurations. Averaging these pixels produces noise cancellation and produces a pixel that is like its original value since noise and picture pixels are highly associated while noise is normally independently and identically distributed.

BACKGROUND

Given that noise is a random phenomenon, it is characterized by a probability density that depicts the noise's intensity distribution. Different types of noises exist, with Gaussian, Salt and Pepper, Speckle, and Poisson being the most common.

Data drop noise, often known as "salt and pepper noise," occurs when some pixels' values are altered to either black or white in the image (black or white dots). Normally, we refer to picture noise with black dots as "pepper noise" and noise with white dots as "salt noise." Typically, mistakes in data transfer, memory cell breakdown, or issues with analog-to-digital converters are to blame for this noise.

The grey levels in digital photographs are disturbed by Gaussian Noise. For this reason, the Gaussian noise model's probability density function or normalization of the histogram with the grey value serve as its primary design and defining features. The random fluctuations in the signal that create it are an idealized version of white noise. The addition of gaussian white noise is the most straightforward method for simulating the impact of noise on a digital image. Each pixel (m,n) of the observation Y is modelled by additive white Gaussian noise (AWGN) as the sum of a pixel from the noiseless image x and a pixel from the noise n:

$$v(i) = u(i) + n(i)$$

where n(i)~N(0,σ2). where v(i) is the observed value, u(i) is the “true” value and

n(i) is the noise perturbation at a pixel i.

Denoising is the process of cleaning up or lowering the image's noise or artifacts. Denoising makes the image more lucid and allows us to plainly notice the image's finer elements. Although it doesn't directly alter the image's brightness or contrast, the elimination of artifacts may make the final image appear brighter. Numerous denoising methods exist, including neighbourhood filtering, anisotropic filtering, total variation reduction, Gaussian filtering, and Non-Local Mean Algorithm.

Method noise is a concept to measure the difference between the non-noisy image and the original image. It is a method for recovering or restoring the raw, unprocessed, or unfiltered elements of the filtered image. A denoising approach must seem like noise even with non-noisy images to be effective, and it should have as little structure as feasible. Using this method noise, the effectiveness of various filtering or smoothing procedures is evaluated by comparing the final product to the original image.

Based on the Gaussian distribution, the Gaussian Smoothing Operator computes a weighted average of the surrounding pixels. It is a realistic model of a defocused lens and is used to eliminate Gaussian noise. To get the desired result, the filter is constructed as an odd sized symmetric kernel (DIP version of a matrix) and passed through each pixel in the region of interest. Sigma is a measure of blurring intensity. The template's size is adjustable using the radius slider. Only for bigger template sizes will large values for sigma result in substantial blurring.

Theorem 1 (Gabor 1960) – The image method noise of the convolution with a gaussian kernel Gh is

$u - G_h * u = -h^2 \Delta u + 0(h^2)$, for h small enough

The Gaussian convolution is optimal in flat parts of the image, but edges and texture are blurred.

Anisotropic filtering is a technique aiming at reducing image noise without removing significant parts of the image content, typically edges, lines or other details that are important for the interpretation of the image. Image u is convolved at x in orthogonal direction Du(x).

$$AF_h u(x) = \int G_h(t) u\left(x + t\frac{Du(x)}{|Du(x)|}\right) dt$$

G is the one-dimensional Gauss function with variance h2

To measure using method noise –

Theorem 2- The image method noise of an anisotropic filter AF_h is

$$u(x) - AF_h u(x) = -\frac{1}{2}h^2 |Du| curv(u)(x) + o(h^2)$$

Where the relation holds when Du(x) ≠ 0 and curv(u)(x) is the curvature

This method noise is zero wherever u behaves locally like a straight line and large in curved edges or texture (where the curvature and gradient operators take high values). As a consequence, the straight edges are well restored while flat and textured regions are degraded.

Total Variation minimization is based on the principle that signals with excessive and possibly spurious detail have high total variation, that is, the integral of the absolute image gradient is high. According to this principle, reducing the total variation of the signal—subject to it being a close match to the original signal—removes unwanted detail whilst preserving important details such as edges.

T V (u) denotes the total variation of u and λ is a given Lagrange multiplier

$$TVF_\lambda(v) = \arg\min TV(u) + \lambda \int | v(x) - u(x)|^2 dx$$

Theorem 3 – The method noise of the Total Variation minimization is

$$u(x) - TVF_\lambda(u)(x) = -\frac{1}{2\lambda} curv(TVF_\lambda(u))(x)$$

However, details and texture can be over smoothed if λ is too small.

Neighborhood filtering restores a pixel by taking an average of the values of neighboring pixels with a similar grey level value.

The problem with these filters is that comparing only grey level values in a single pixel is not so robust when these values are noisy. Hence nonlocal means was proposed.

Spatial domain filtering is classified into linear and nonlinear filters. Non-Local means filter is one of the spatial domain filters. A single pixel is recovered by averaging all observed pixels in non-local means filtering. Many changes are performed in the original non-local means filter to improve the performance. So the estimated value for pixel i –

$$NL[v](i) = \sum_{j€I} w(i,j)v(j)$$

Where, $v = \{\, v(i) | i\epsilon\, I\}$

It adjusts each pixel value with a weighted average of other pixels whose neighborhood has a similar geometrical configuration. Since image pixels are highly correlated while noise is typically independently and identically distributed, averaging of these pixels results in noise cancellation and yields a pixel that is like its original value. This results in much greater post-filtering clarity, and less loss of detail in the image compared with local mean algorithms. If compared with other well-known denoising techniques, non-local means adds "method noise" (i.e. error in the denoising process) which looks more like white noise, which is desirable because it is typically less disturbing in the denoised product. Recently non-local means has been extended to other image processing applications such as deinterlacing, view interpolation, and depth maps regularization.

METHODOLOGY

The broad idea of the Non-Local Means (NLM) algorithm is to estimate the value of actual pixel in the denoised image using a weighted mean of pixel neighborhoods having some similarity in the noisy image. Preferably pixel resemblance should be explored in the complete image but for high resolution images, this procedure can take quite some time. So, pixel similarity is searched only in some neighborhood of each pixel, defined by the variable big_window. Also, to increase the performance while calculating the similarity of each pixel, instead of considering only one pixel, a small area of pixels is chosen around it, given by the small_window parameter.

On the whole we need to follow the following basic steps:

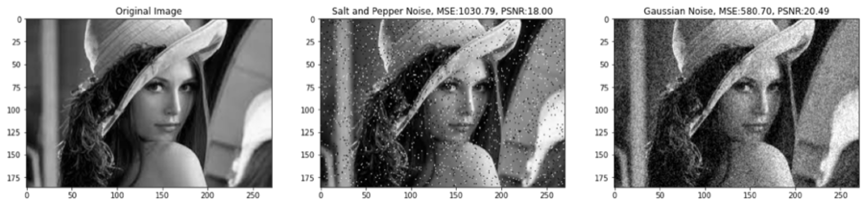
1. We take a sample image on which we want to test our algorithm on , and add gaussian and salt and pepper noise to the image to make it a noisy image. The resulting two images (one with salt and pepper noise and other with gaussian noise) will act as our base images on which we will perform denoising.
2. The main goal of the Non-Local Means (NLM) algorithm is to estimate the actual pixel value using a weighted mean of pixel neighbourhoods having some similarity in the noisy image. Given a discrete noisy image $v = \{v(i) \mid i \in I\}$, the estimated value NL[v](i), for a pixel i, is computed as a weighted average of all the pixels in the image,
   $\mathrm{NL}[v](i) = \sum_j w(i,j)v(j)$
   where the family of weights $\{w(i,j)\}j$ depend on the similarity between the pixels i and j, and satisfy the usual conditions $0 \leq w(i, j) \leq 1$ and $\sum_j w(i,j)v = 1$
3. Ideally, we need to search the pixel similarity in the entire image, but in high-resolution pictures, it is not always possible to do this. So we will define a window of size MxM.
4. To increase the performance while calculating the similarity of each pixel, instead of considering only one pixel, a small area of pixels is chosen around it, given by the NxN parameter.
5. We will also have another parameter h. This is the filtering parameter that determines the impact of each exponential value in the calculations, i.e. the impact of the similarities as a function of distance. It is used to control the weight of the patch difference values.
6. Now that we have our parameters M, N, and h, we can feed them to our NLM algorithm. Each pixel is traversed in the calculated big window MxM neighbourhood. For each such pixel, we get a small window NxN patch and the squared norm of this 2D window and the current pixel's 2D window are taken.
7. This norm is multiplied by -1/(size of neighbourhood window) and its exponential is taken. This value is used to calculate the numerator of the expected NLMeans value for each pixel (as defined in the paper given to us). The denominator of this fraction is calculated by the summation of such exponential terms.
8. The parameters used here were the default parameters as used in the paper. In case of salt and pepper noise, h was taken as 36 and for gaussian noise h was taken as 27. The window sizes big window is taken as 21 and small window is taken as 7.

RESULTS

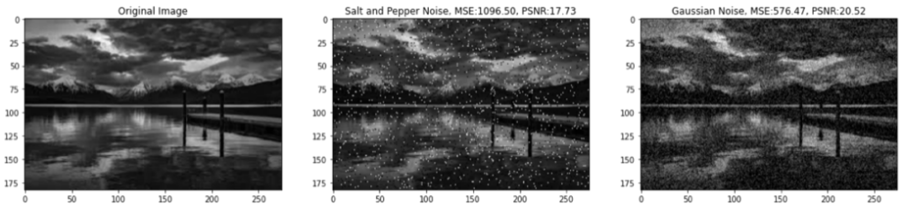
The algorithm defined above was tried on 5 different images and the results can be seen below:

As an initial step, we added salt and pepper and gaussian noise to each of these 5 images.

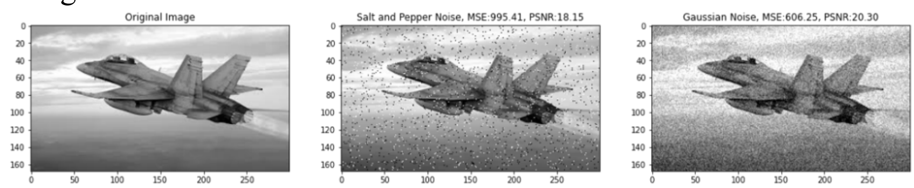
- Image 1 with noise:



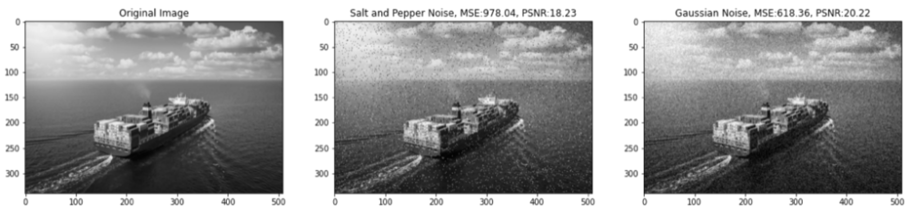
- Image 2 with noise:



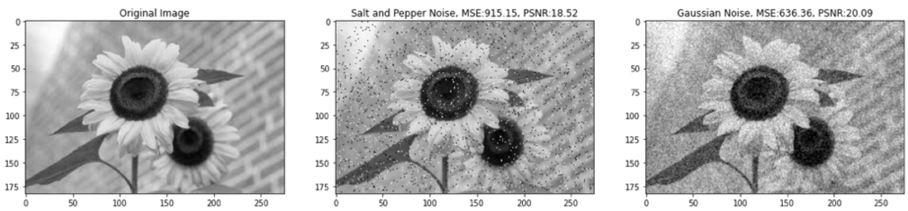
- Image 3 with noise:



- Image 4 with noise:

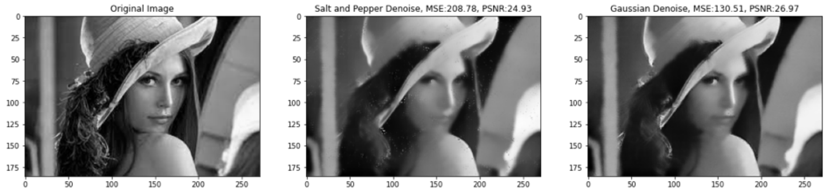


- Image 5 with noise

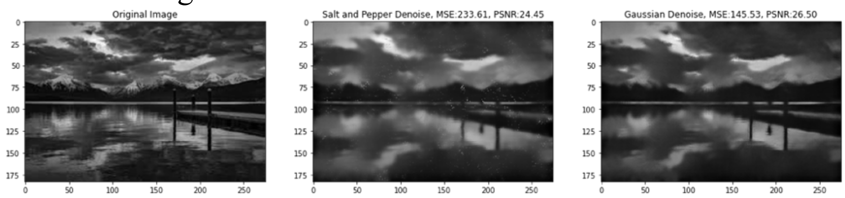


On running the denoising NLM algorithm on the noisy images, we got the following results:

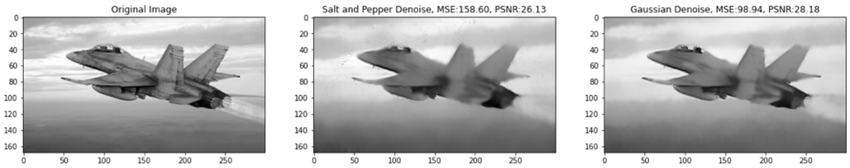
- Denoised image 1



- Denoised image 2

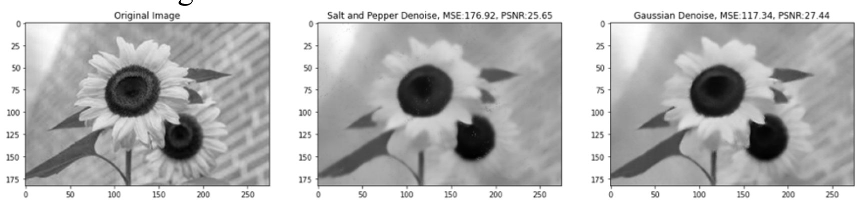


- Denoised image 3



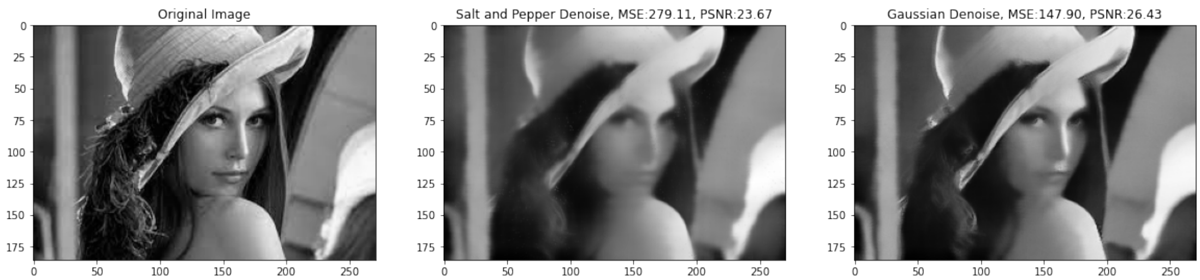
- Denoised image 4

- Denoised image 5



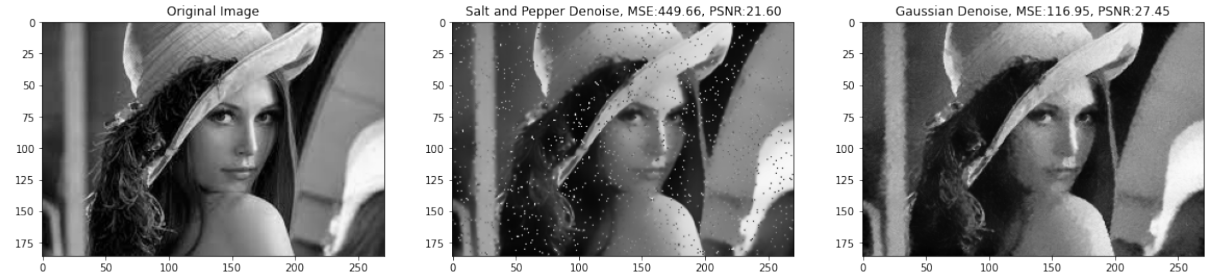
We also tried experimenting with different parameters to observe the changes in the results:

1. Image 1 with h=50 for gaussian noise and h=30 for salt and pepper noise

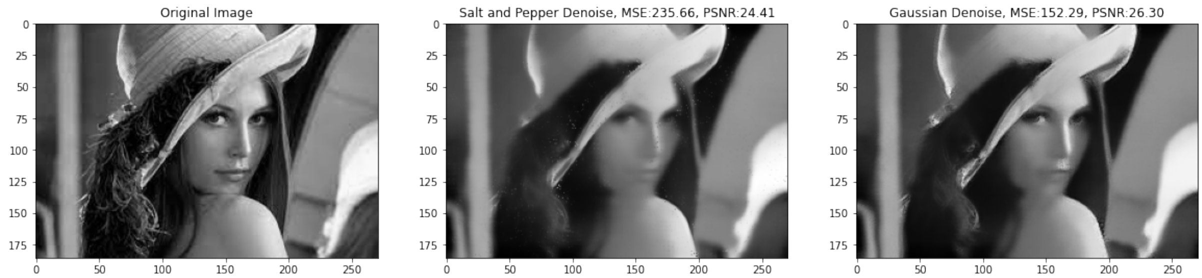


Compared to our previous results we notice there is more blur in the resultant image.

2. On changing the small window size from 7 to 3 and big window size from 21 to 14 , we see the following results. As can be seen from both the resultant image and quantitative measures, the quality of denoising has increased on decreasing the window size.



3. On changing the small window size from 7 to 9 and big window size from 21 to 27 we obtain the following results. We notice that the quality of denoising has slightly deteriorated.



Conclusion:

1. In this project report we have taken 5 different images and added salt and pepper and gaussian noise to it. On adding the noise, we notice the degradation in the image quality.

After adding noise, we also quantitively examined the qualities of images using PSNR and MSE

Gaussian Noise					
	Image 1	Image 2	Image 3	Image 4	Image 5
PSNR	20.47	20.52	20.3	20.22	20.09
MSE	583.57	576.47	606.99	618.36	636.36

Salt and Pepper Noise					
	Image 1	Image 2	Image 3	Image 4	Image 5
PSNR	17.92	17.73	18.1	18.23	18.52
MSE	1050.72	1096.5	1005.98	978.04	915.15

2. We then apply our NLM implementation to each of the noisy images. We notice that NLM performs better on removing Gaussian noise than salt and pepper noise. We also quantitatively assess the results as can be seen below:

Gaussian Noise					
	Image 1	Image 2	Image 3	Image 4	Image 5
PSNR	26.97	26.5	28.18	26.89	27.5
MSE	130.51	145.53	98.94	133.18	115.72

Salt and Pepper Noise					
	Image 1	Image 2	Image 3	Image 4	Image 5
PSNR	24.93	24.45	26.13	25.01	25.63
MSE	208.78	233.61	158.6	205.35	177.78

3. Thirdly we play around with the parameters and notice that
    a. On increasing the h value, results in noisier image after denoising.
    b. On decreasing the small and big window size from the one taken in the research paper , we notice increase in the denoising quality
    c. On increasing the small and big window size from the one taken in the research paper , we notice decrease in the denoising quality

4. REFLECTION

1. One of the main problems that we faced was due to the time taken to denoise the image. Since we were operating our code on CPU and not GPU , considerable time was taken to denoise each image.
2. We notice that NLM seems to be doing a better job at denoising gaussian noise than salt and pepper noise. To remove salt and pepper noise we might need to consider another method.
3. NLM requires the manual tuning of hyperparameters like window sizes, h , sigma values, etc. These values are critical and effect the denoising quality . Since we are currently manually setting these parameters , it is prone to error. A lot of the current state of the art machine learning algorithms incorporate automated parameter tuning where the algorithm itself learns the best parameters based on previous data.

REFERENCES

1) Wilson, B. and Dhas, D.J.P.M. (2013) *A survey of non-local means based filters for image denoising*, *International Journal of Engineering Research & Technology*. IJERT-International Journal of Engineering Research & Technology. Available at: https://www.ijert.org/a-survey-of-non-local-means-based-filters-for-image-denoising (Accessed: December 6, 2022).
2) *Image noise reduction and filtering techniques - IJSR* (no date). Available at: https://www.ijsr.net/archive/v6i3/25031706.pdf (Accessed: December 7, 2022).
3) Fan, L. *et al.* (2019) *Brief review of image denoising techniques - visual computing for Industry, Biomedicine, and art*, *SpringerOpen*. Springer Singapore. Available at: https://vciba.springeropen.com/articles/10.1186/s42492-019-0016-7 (Accessed: December 6, 2022).
4) Mantri, N. (2019) *Image denoising and various image processing techniques for it*, *OpenGenus IQ: Computing Expertise & Legacy*. OpenGenus IQ: Computing Expertise & Legacy. Available at: https://iq.opengenus.org/image-denoising-and-image-processing-techniques/ (Accessed: December 6, 2022).
5) *Basics of Image Processing* (no date) *Denoising - Basics of Image Processing*. Available at: https://vincmazet.github.io/bip/restoration/denoising.html (Accessed: December 6, 2022).
6) *Basics of Image Processing* (no date) *Denoising - Basics of Image Processing*. Available at: https://vincmazet.github.io/bip/restoration/denoising.html (Accessed: December 6, 2022).
7) Buades, A., Coll, B. and Morel, J.-M. (2011) *Non-local means denoising*, *Image Processing On Line*. Available at: https://edit.ipol.im/pub/art/2011/bcm_nlm/ (Accessed: December 6, 2022).
8) *What is method noise?* (2020) *Quick Image Processing Research Guide*. Available at: https://www.gofastresearch.com/2020/04/what-is-method-noise.html (Accessed: December 6, 2022).
9) *Computer Vision Demonstration Website* (no date) *Gaussian Filtering - Computer Vision Website Header*. Available at: https://www.southampton.ac.uk/~msn/book/new_demo/gaussian/ (Accessed: December 6, 2022).