CSC831 – Assignment Part 3 - Report

Max Abimbola
210069943

Table of content

Introduction

This assignment focused on the common image processing problem, image denoising. This problem concerns the improvement of image quality (removing noise) while preserving or improving image details. One of the most popular methods for this is the DnCNN model introduced in 2016 by Zhang, K., Zuo, et al. This method sees the use of a CNN to denoise images. In this assignment I will be using several image denoising methods to improve the quality of 25 images each grouped into sets of increasingly poor image quality. To test the effectiveness of my implementations I will then be using several, common evaluation metrics such as MSE, SSIM and PSNR.

My Algorithm

Description

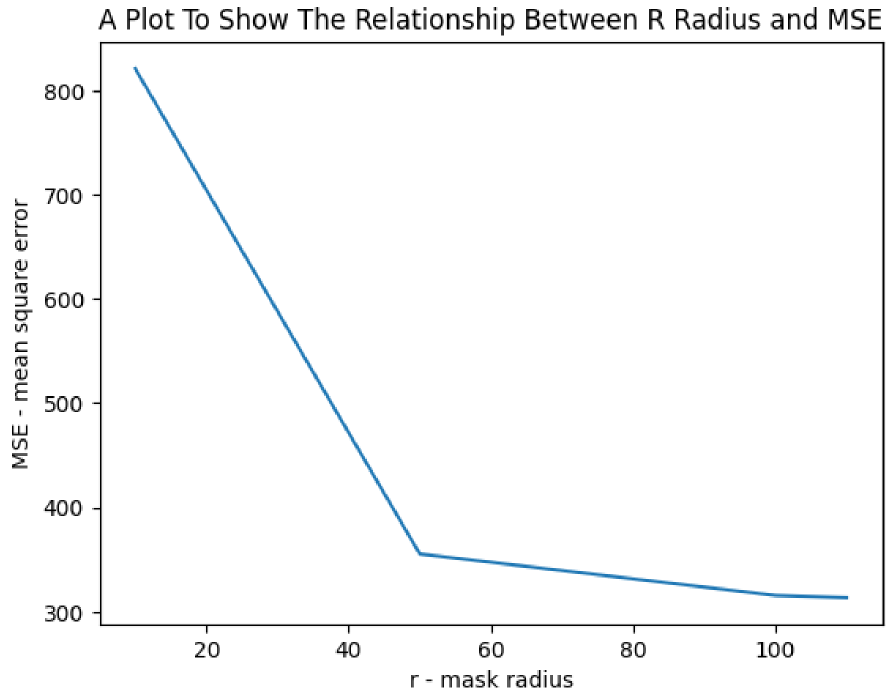
The denoising algorithm begins by reading a noisy image, applying Fourier Transform to each channel (after splitting). This is so I can apply the mask to each individual filter as noise frequency isn't equal in each channel.I then create a frequency domain mask to filter out high-frequency noise. The Inverse Fourier Transform is then applied, followed by clipping to ensure pixel values are in the valid range. Subsequently, Gaussian and bilateral filters are employed to enhance denoising, with the bilateral filter contributing to edge preservation. The algorithm accommodates user-defined parameters, such as the radius for noise filtering 'r'. [2]

Results

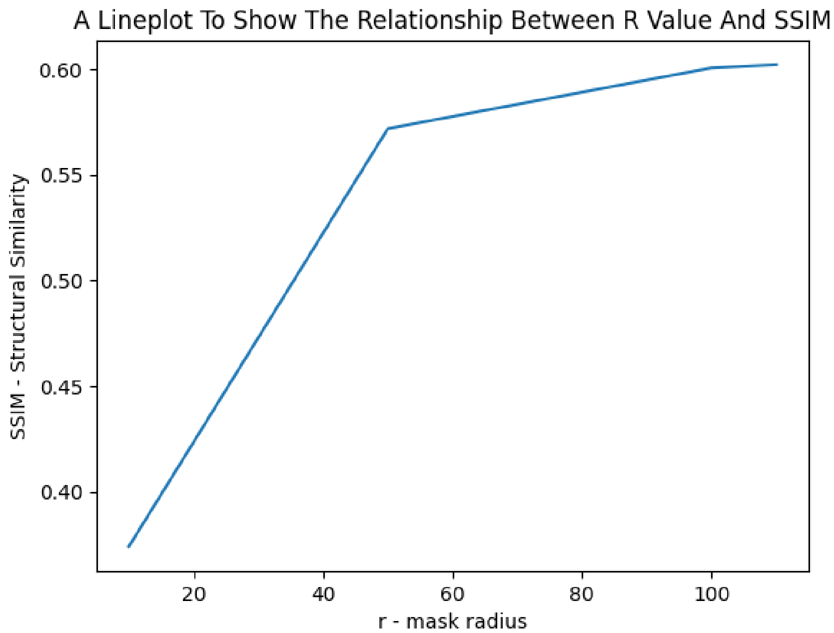
'R' value optimisation
The radius of the mask is instrumental to the performance of the algorithm, so I wrote a script to find the optimum value for R based on evaluation metrics (MSE, SSIM, PSNR)

The results showed that "110" was the best R value for all three metrics

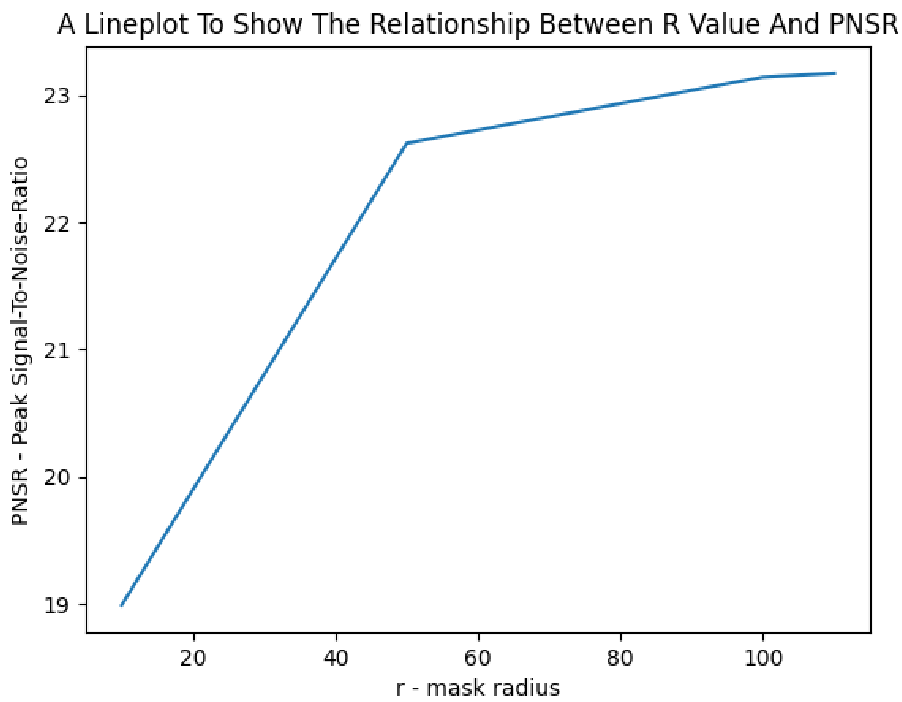
MSE



SSIM



PNSR



The results of me denoising using this algorithm are as follows [3]

MSE was reduced from **2049 - 313.44**
SSIM was increased from **0.35 - 0.6**
PSNR was increased from **15.01 - 23.17**

These results show a very large numerical increase in quality and aesthetically the image looks a lot better once denoised as well.

Key findings from other algorithms

In the assignment I also implemented 3 other denoising algorithms and compared their results to those from my custom one.

Median Filter [3]

- Significant reduction in MSE, increase in SSIM and PSNR
- Median filter is effective at reducing noise by replacing each pixel with the median of its neighbouring pixels, this reserves edges well leading to improved score

MSE was reduced from **2049 - 519.56**
SSIM was increased from **0.35 - 0.53**

PSNR was increased from **15.01 - 20.97**

Mean Filter

[4]

- SImilar trends as the median filter, with further improvements in evaluation metrics
- Mean filter smoothens the picture by replacing each pixel with the average of its neighbours. I

MSE was reduced from **2049 - 223.43**
SSIM was increased from **0.35 - 0.7**
PSNRwas increased from **15.01 - 24.64**

Wavelet denoising

[5]
Bayes - Wavelet denoising using the BayesShrink method, is well-suited for images with salt-and-pepper noise (as the image is). It operates in the wavelet domain, where noise often manifests as high-frequency components. BayesShrink thresholding effectively suppresses these high-frequency noise components while retaining important image features, contributing to improved SSIM and PSNR.

MSE was reduced from **2049 - 291.63**
SSIM was increased from **0.35 - 0.6**
PSNR was increased from **15.01 - 23.48**

[6]
Visushrink - Wavelet denoising using Visushrink method may be less-suited for denoising salt and pepper noise as it assumes the noise is in a gaussian distribution. This paired with its thresholding operation, which also works under the assumption of gaussian noise, explains the reduction in denoising quality

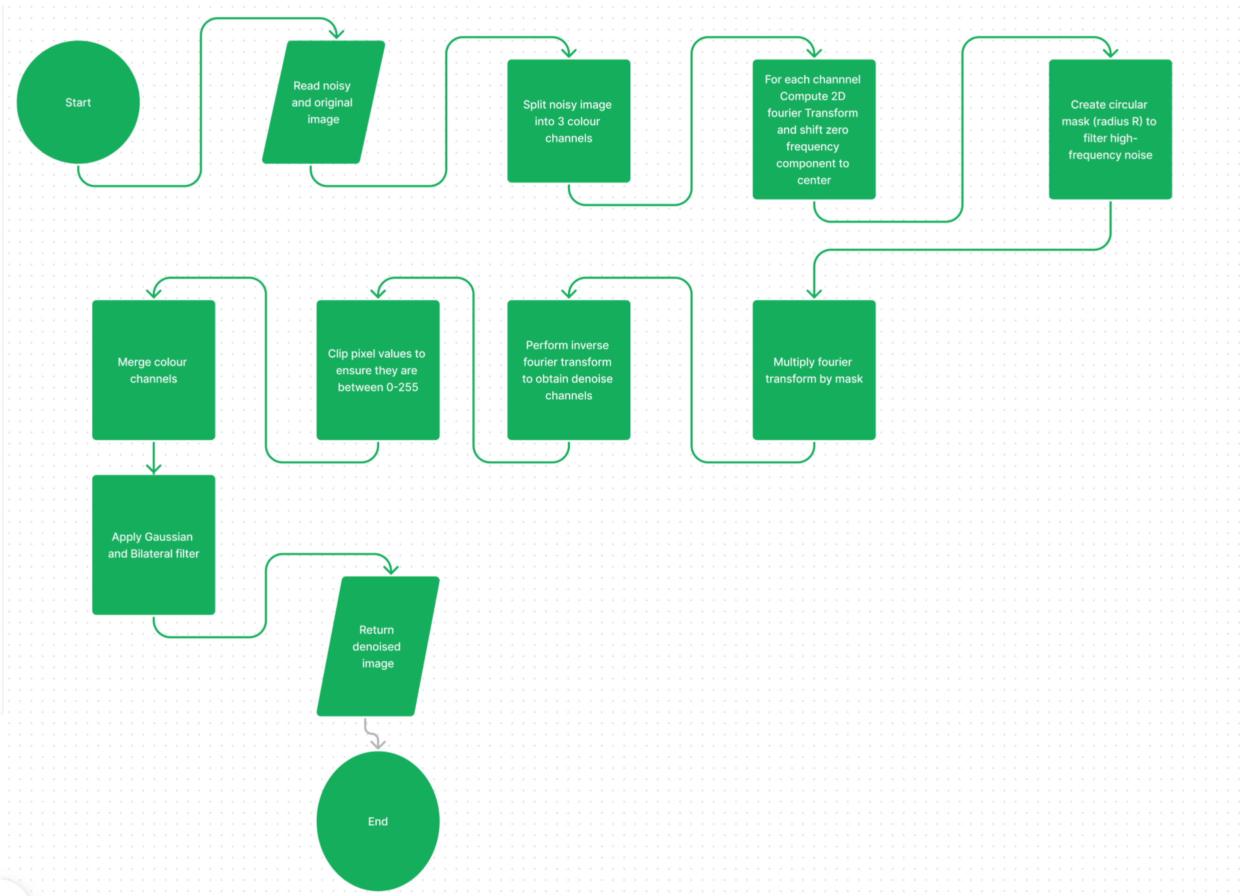
MSE was reduced from **2049 - 641.43**
SSIM was increased from **0.35 - 0.42**
PSNR was increased from **15.01 - 20.06**

Conclusions

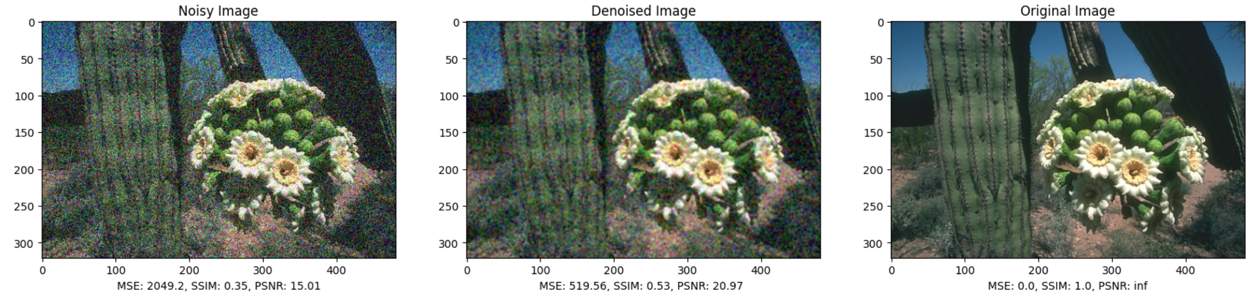
In conclusion, the custom denoising algorithm, optimised with an 'r' value of 110, demonstrated substantial improvements in MSE, SSIM, and PSNR, significantly enhancing image quality. Comparative analysis with other algorithms, such as Median and Mean Filters, highlighted the effectiveness of each method in addressing specific noise types.

Reference list

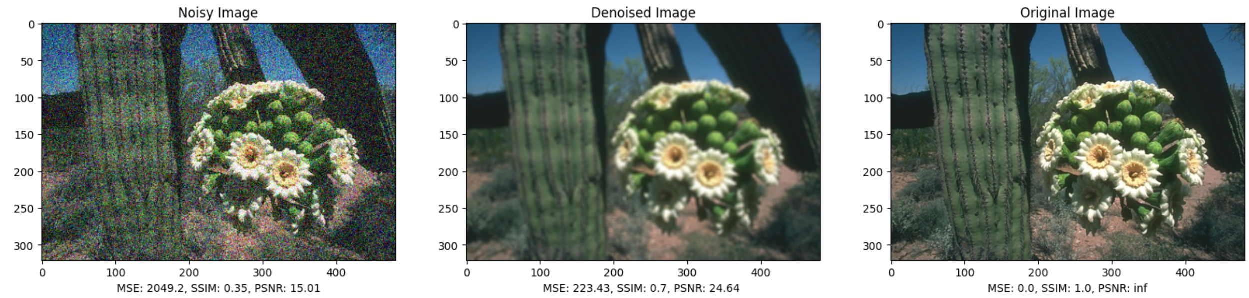
[1]
Zhang, K., Zuo, W., Chen, Y., Meng, D. and Zhang, L. (2017). Beyond a Gaussian Denoiser: Residual Learning of Deep CNN for Image Denoising. *IEEE Transactions on Image Processing*, 26(7), pp.3142–3155. doi:https://doi.org/10.1109/tip.2017.2662206.
[2]



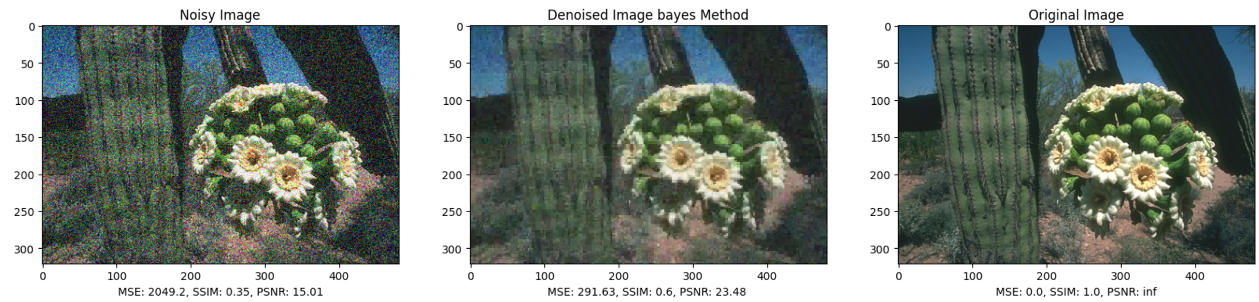
[3] - *Image denoising of noisy50/0004.png*


Noisy Image
Denoised Image
Original Image
MSE: 2049.2, SSIM: 0.35, PSNR: 15.01
MSE: 519.56, SSIM: 0.53, PSNR: 20.97
MSE: 0.0, SSIM: 1.0, PSNR: inf

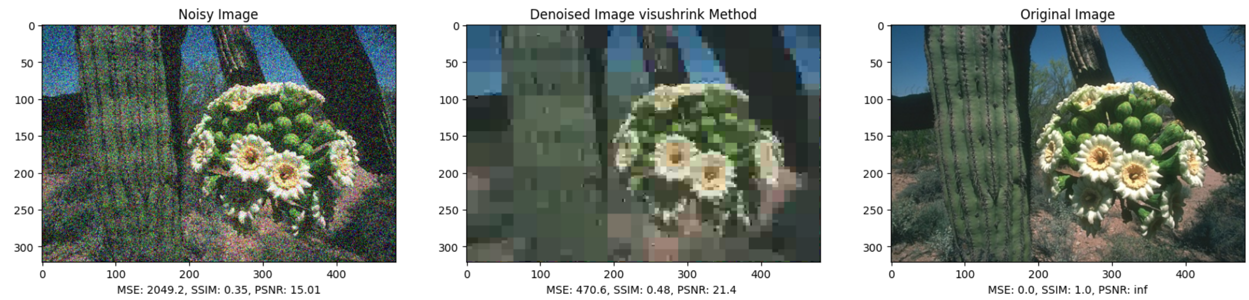
[4]


Noisy Image
Denoised Image
Original Image
MSE: 2049.2, SSIM: 0.35, PSNR: 15.01
MSE: 223.43, SSIM: 0.7, PSNR: 24.64
MSE: 0.0, SSIM: 1.0, PSNR: inf

[5]


Noisy Image
Denoised Image bayes Method
Original Image
MSE: 2049.2, SSIM: 0.35, PSNR: 15.01
MSE: 291.63, SSIM: 0.6, PSNR: 23.48
MSE: 0.0, SSIM: 1.0, PSNR: inf

[6]


Noisy Image
Denoised Image visushrink Method
Original Image
MSE: 2049.2, SSIM: 0.35, PSNR: 15.01
MSE: 470.6, SSIM: 0.48, PSNR: 21.4
MSE: 0.0, SSIM: 1.0, PSNR: inf