Image Denoising

Homework n°2 - Experimental report

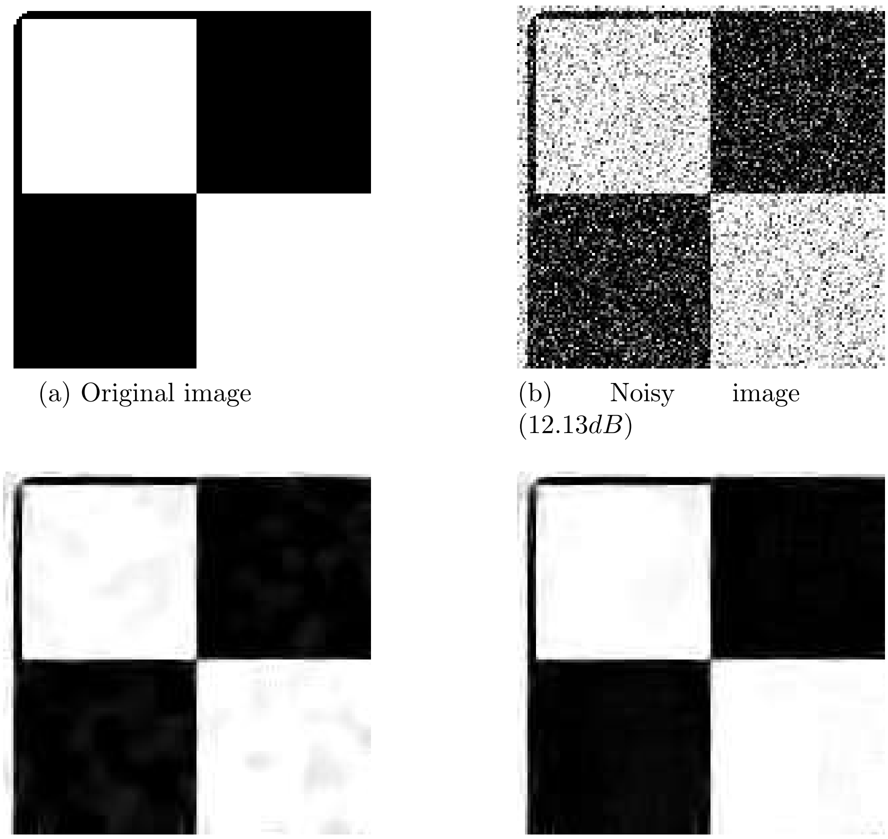
1 Multi-Scale DCT denoising

The DCT algorithm enables to denoise an image. However, it suffers from some drawbacks, one of them being that the low-frequencies are not handled properly, because of the limited size of the considered patches.

A possible improvement is the Multi-Scale DCT (MS DCT) algorithm, introduced in [2].
The main idea of this algorithm is to take the noisy image, and create several versions of this image with different scales. Then, the DCT denoising algorithm is applied to each of these images, and the denoised images are combined to create the final denoised image.

1.1 Basic experiments

In Figure 1, we compare the results obtained with DCT and MS DCT on a very basic image, for $\sigma = 90$, DCT size $= 8$, scales $= 5$, $f_{\text{rec}} = 0.4$.



(a) Original image

(b) Noisy image ($12.13dB$)

(c) DCT ($26.85dB$)

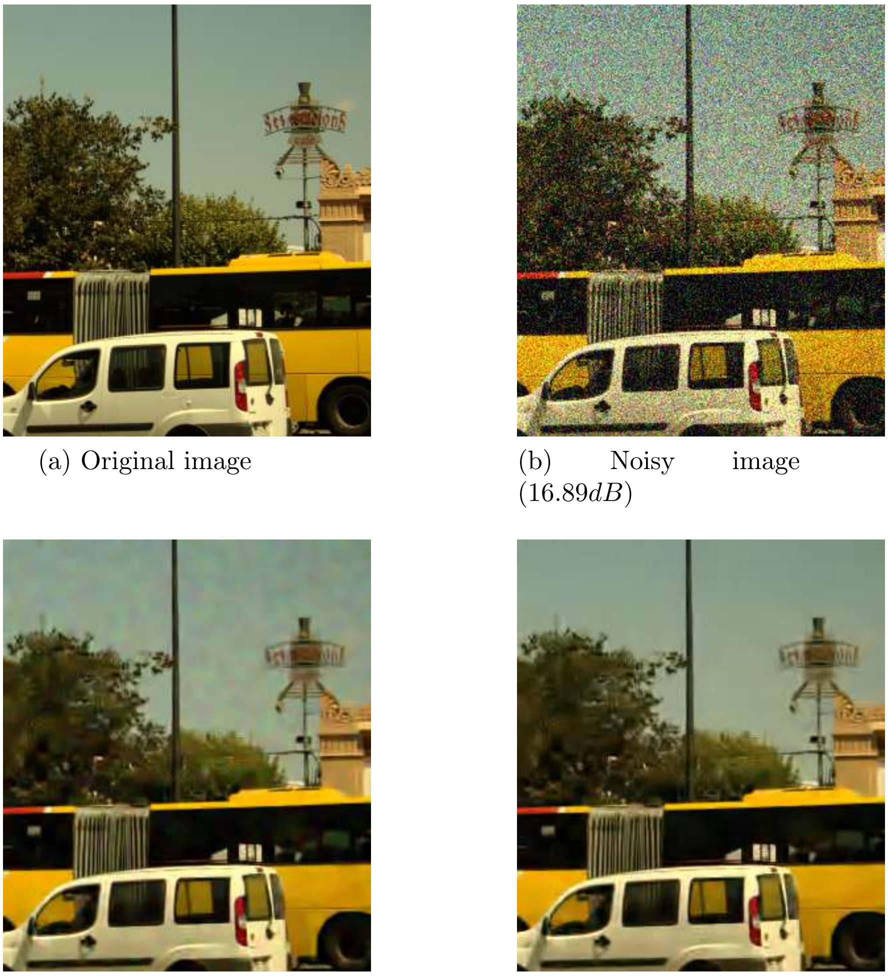
(d) MS DCT ($27.79dB$)

Figure 1: Comparison between one-scale DCT and multi-scale DCT (chessboard)

We see that MS DCT enables to get rid of many artifacts present with DCT, especially in the

uniform patches, where there are no edges nor textures. This is what we explained above: MS DCT handles low-frequencies quite well compared to DCT.
Some artifacts still remain (for instance near the edges), but much less.
We also see that the PSNR increases by about $1dB$ when using MS DCT, which proves that the obtained image is overall better.
So the problem present with DCT, which was to handle low-frequencies, seems to have been mostly solved.

Let us use a more complex picture, this time with $\sigma = 40$, DCT size $= 8$, scales $= 5$, $f_{\text{rec}} = 0.4$, in Figure 2.



(a) Original image

(b) Noisy image ($16.89dB$)

(c) DCT ($26.93dB$)

(d) MS DCT ($27.06dB$)

Figure 2: Comparison between one-scale DCT and multi-scale DCT (traffic)

This time, the difference in terms of PSNR between DCT denoising and MS DCT denoising seems rather small. However, when looking at the picture, the difference is clearly visible, especially (once again) in uniform areas, that is, the sky. However the image remains blurry, as can be seen from the text in the top right corner which, even though it is readable in the original image, is not for any of the denoised images.

(a) No adaptive aggregation ($27.03dB$)

(b) Adaptive aggregation ($27.06dB$)

Figure 3: Comparison between one-scale DCT and multi-scale DCT (traffic)

1.2 Impact of the parameters

1.2.1 Adaptive aggregation

Figure 3 shows the impact of adaptive aggregation: though the PSNR does not change much, we notice a difference near the edges, where some artifacts disappear thanks to adaptive aggregation.

1.2.2 Size of the patches

Figure 4 shows the impact of the patch size on the difference between DCT and MS DCT: when using a large patch size, DCT manages low frequencies a better way, and the difference between DCT and MS DCT reduces.
Note that the results in the figure were found by averaging 5 values each time.
And the larger the patch size is, the better DCT handles uniform areas.

1.2.3 Impact of f_{rec}

Figure 5 shows the impact of f_{rec} for MS DCT: when $f_{rec} = 1$, Gibbs oscillations remain, unlike lower values of f_{rec}. This is because using $f_{rec} < 1$ introduces Gibbs oscillations on the different scales, which can then compensate each other when the images are recombined.

Patch size	**DCT**	**MS DCT**
4	$26.30\,dB$	$26.84\,dB$
8	$26.93\,dB$	$27.06\,dB$
12	$26.89\,dB$	$26.88\,dB$
16	$26.79\,dB$	$26.73\,dB$

Figure 4: Impact of the patch size

(a) $f_{rec} = 0.4$ (b) $f_{rec} = 1$

Figure 5: Impact of f_{rec}

1.2.4 Number of scales

Figure 6 shows the impact of the number of scales. We see that for 2 scales, though the sky is already better denoised than with one scale, the colors are still not perfectly restablished. However, for 7 scales, the result is better.
This could be expected from what we said better: the more scales we use, the better low frequencies are handled, the better the uniform areas are denoised.

1.2.5 Impact of σ

Figure 7 shows the impact of σ on the difference between DCT and MS DCT: the higher σ is, the better MS DCT is compared to DCT. And of course, both algorithms perform better for low values of σ...

(a) 2 scales

(b) 7 scales

Figure 6: Impact of the number of scales

σ	**DCT**	**MS DCT**
10	$34.37\,dB$	$34.39\,dB$
40	$26.93\,dB$	$27.06\,dB$
100	$23.11\,dB$	$23.60\,dB$

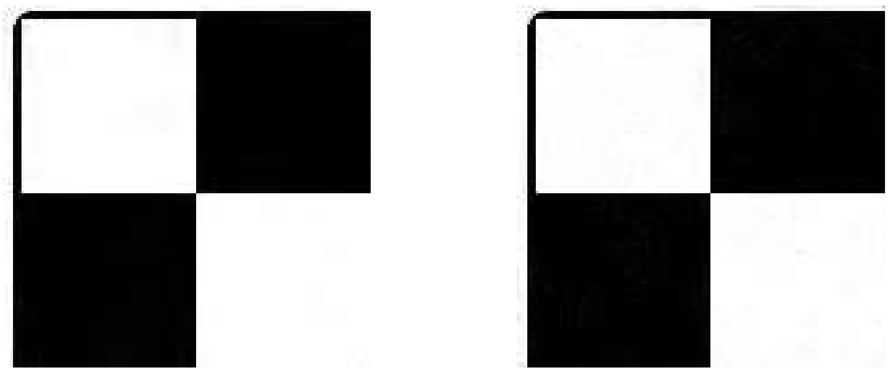
Figure 7: Impact of σ

2 Non-Local Means

Non-Local Means (NLM) Denoising [1] is a denoising technique that differs from DCT in the sense that no Fourier nor Cosine transform is used: this algorithm does not explicitly manipulate frequencies.

Instead, for a given pixel, the idea is to consider a patch around this pixel, and to find similar patches in the image (in a ball centered at the pixel, and of given radius). Then, the denoised patch is obtained by averaging over the similar patches found previously, with weights depending on the L^2 norm between the patches. Finally, since for a given pixel i several patches containing i have been denoised, the value of i is obtained by averaging the value of i for the different denoised patches.

Here, the only parameter we have access to is the standard deviation of the noise σ.
Let us directly compare the results of MS DCT and NLM.



(a) MS DCT ($39.88dB$) (b) NLM ($36.53dB$)

Figure 8: MS DCT and NLM for a chessboard, $\sigma = 30$

We observe in Figure 8 that in the image denoised by NLM, some grey pixels remain in the white areas of the chessboard. This is because the pixels are denoised using an average of pixels (though this is done at the scale of a patch), and though this averaging process reduces the noise (by reducing its variance), it does not entirely remove it.
MS DCT however will remove the frequencies with low amplitude assumed to be noise, thus getting rid of these grey pixels.
In this particular case, the PSNR is much better by using MS DCT than NLM (which is still the case for an entire chessboard, with a difference of more than $2\,dB$).

In Figure 9, we see that for NLM, some details can be lost, or attenuated. For instance, some of the white lines on a road are thinner, and the color of the road is more homogeneous than in the original picture. We see in Figure 10 that for large values of σ, this is even more visible. This "attenuation problem" can once again directly be understood as a consequence of the averaging process (for the road, the part that has a lighter color is initially already thin, so there are not many patches containing this color).

Moreover, some horizontal lines appear for NLM (as can be seen in the sky in Figure 11).

(a) MS DCT ($28.36dB$)

(b) NLM ($27.69dB$)

Figure 9: MS DCT and NLM for a real-world image, $\sigma = 30$

Figure 10: Image denoised with NLM, $\sigma = 60$

Figure 11: Artifacts for NLM in uniform areas

3 Conclusion

From what we have seen before, the main advantage of MS DCT compared to DCT is the way the low frequencies are handled, resulting in better denoised uniform areas.
However, when using large enough patch sizes, DCT also manages to handle correctly low frequencies, and the advantage of MS DCT is then less clear.
As for the other parameters, f_{rec} must not be chosen too high, the number of scales should not be too low, and adaptive aggregation should be used. This helps reduce artifacts and ringing. However, for the number of scales, this is a trade-off between good denoising and computation time.

NLM has the advantage of not having Gibbs oscillation, since it does not directly work on frequencies. It also has the advantage of not requiring many parameters, and to be very intuitive. However, it tends to make images less intense (because of the averaging process), some details disappear, and in homogeneous areas some noise may persist (like with the chessboard, or the horizontal lines seen in Figure 11).

Overall, NLM is an easier algorithm than MS DCT, but which tends to achieve worse results. However, for small enough values of σ, it can achieve results comparable to MS DCT.

References

[1] Antoni Buades, Bartomeu Coll, and Jean-Michel Morel. Non-Local Means Denoising. *Image Processing On Line*, 1:208–212, 2011. https://doi.org/10.5201/ipol.2011.bcm_nlm.

[2] Nicola Pierazzo, Jean-Michel Morel, and Gabriele Facciolo. Multi-Scale DCT Denoising. *Image Processing On Line*, 7:288–308, 2017. https://doi.org/10.5201/ipol.2017.201.