BM3D : PSNR calculation

2023.01.16.

BM3D : PSNR calculation

TABLE III

GRAYSCALE-IMAGE DENOISING: OUTPUT PSNR (DB) OF THE PROPOSED BM3D ALGORITHM.

σ / PSNR	*C.man* 256^2	*House* 256^2	*Peppers* 256^2	*Lena* 512^2	*Barbara* 512^2
10 / 28.14	34.18	36.71	34.68	35.93	34.98
25 / 20.18	29.45	32.86	30.16	32.08	30.72
35 / 17.25	27.93	31.38	28.52	30.56	28.98
75 / 10.63	24.05	27.20	24.48	27.02	25.10
100 / 8.14	22.81	25.50	22.91	25.57	23.49

calculate PSNR for C.man, House, Lena and Barbara
for σ = 10, 25, 35, 75, 100

BM3D : First evaluation

calculate PSNR for C.man, House, Lena and Barbara
for σ = 10, 25, 35, 75, 100

	C.man	**House**	**Lena**	**Barbara**
10	21.66	22.45	25.54	26.29
25	21.47	22.04	24.96	25.98
35	20.96	22.00	24.43	25.03
75	18.30	18.91	21.78	21.29
100	16.34	16.49	17.77	17.65

σ / PSNR	*C.man* 256^2	*House* 256^2	*Lena* 512^2	*Barbara* 512^2
10 / 28.14	34.18	36.71	35.93	34.98
25 / 20.18	29.45	32.86	32.08	30.72
35 / 17.25	27.93	31.38	30.56	28.98
75 / 10.63	24.05	27.20	27.02	25.10
100 / 8.14	22.81	25.50	25.57	23.49

σ = 10 σ = 25 σ = 35 σ = 75 σ = 100

BM3D : Kaiser Window

convolution 과정에서 발생하는 border distortion?

Reduce the border effects.

- Use a $N_1 \times N_1$ Kaiser window (with parameter β) as part of the weights in (12) in order to reduce border effects which can appear when certain 2D transforms (e.g. the 2D DCT, the 2D DFT, or periodized wavelets) are used.

calculate PSNR for C.man, House, Lena and Barbara
for σ = 10, 25, 35, 75, 100

	C.man	House	Lena	Barbara
10	24.45	26.14	27.82	26.58
25	24.16	25.90	27.44	26.05
35	23.50	25.43	26.68	24.41
75	21.12	22.70	23.52	21.14
100	17.87	18.04	18.31	17.69

σ / PSNR	*C.man* 256^2	*House* 256^2	*Lena* 512^2	*Barbara* 512^2
10 / 28.14	34.18	36.71	35.93	34.98
25 / 20.18	29.45	32.86	32.08	30.72
35 / 17.25	27.93	31.38	30.56	28.98
75 / 10.63	24.05	27.20	27.02	25.10
100 / 8.14	22.81	25.50	25.57	23.49

σ = 10 σ = 25 σ = 35 σ = 75 σ = 100

BM3D : Modify Parameters

- Normal Profile. This profile offers a reasonable compromise between computational complexity and denoising performance. It is divided in two cases depending on the level of noise:

$\sigma \leq 40$ the noise is not too severe to affect the correctness of the grouping, hence the thresholding in the d-distance (4) is disabled by setting $\lambda_{2D} = 0$ and relatively small block sizes are used, $N_1^{ht}, N_1^{wie} = 8$.

$\sigma > 40$ corresponds to high level of noise, hence $\lambda_{2D} = 2$ is used to improve the correctness of the grouping and larger block sizes are used, $N_1^{ht} = 12$ and $N_1^{wie} = 11$.

calculate PSNR for C.man, House, Lena and Barbara
for σ = 10, 25, 35, 75, 100

	C.man	House	Lena	Barbara
10	24.58	26.14	27.81	26.64
25	24.38	25.90	27.43	26.24
35	23.85	25.35	26.67	24.96
75	16.54	15.68	16.96	16.74
100	12.49	12.56	12.60	12.72

σ / PSNR	*C.man* 256^2	*House* 256^2	*Lena* 512^2	*Barbara* 512^2
10 / 28.14	34.18	36.71	35.93	34.98
25 / 20.18	29.45	32.86	32.08	30.72
35 / 17.25	27.93	31.38	30.56	28.98
75 / 10.63	24.05	27.20	27.02	25.10
100 / 8.14	22.81	25.50	25.57	23.49

σ = 10 σ = 25 σ = 35 σ = 75 σ = 100

σ = 10 σ = 25 σ = 35 σ = 75 σ = 100

σ = 10 σ = 25 σ = 35 σ = 75 σ = 100

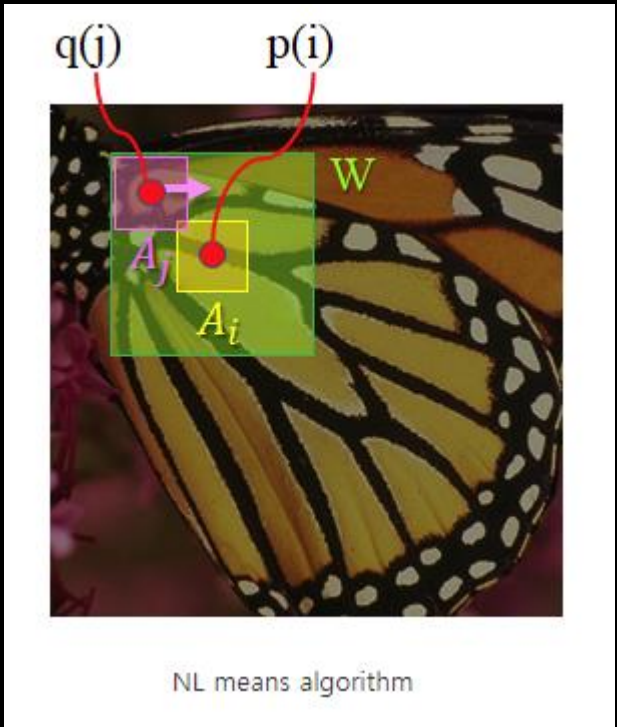
σ = 10 σ = 25 σ = 35 σ = 75 σ = 100

NL algorithm for image denoising : Paper reading

2023.01.16.

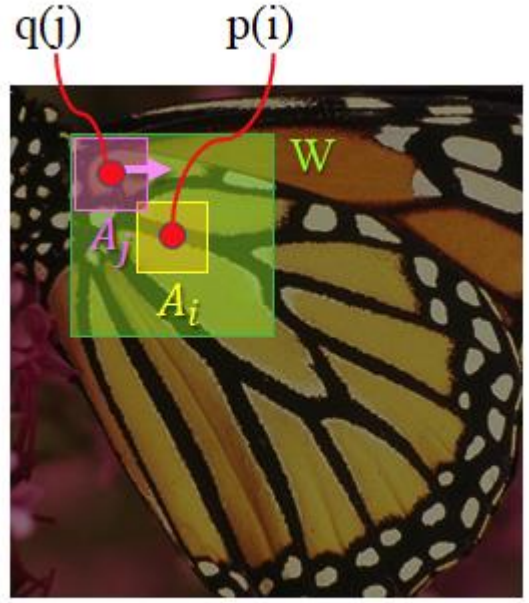
NL algorithm

w(p, q1), w(p, q2) > w(p, q3)



NL means algorithm

NL algorithm



NL means algorithm

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

$$u(p) = \frac{1}{C(p)} \int_\Omega v(q) f(p,q)\, \mathrm{d}q.$$

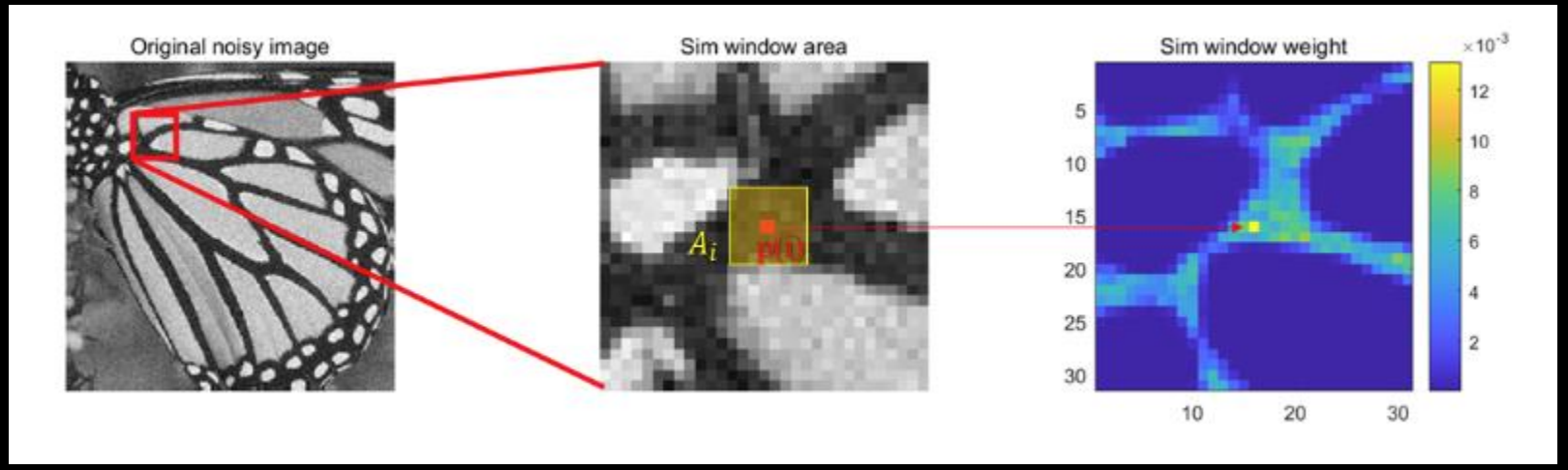
$$C(p) = \int_\Omega f(p,q)\, \mathrm{d}q.$$

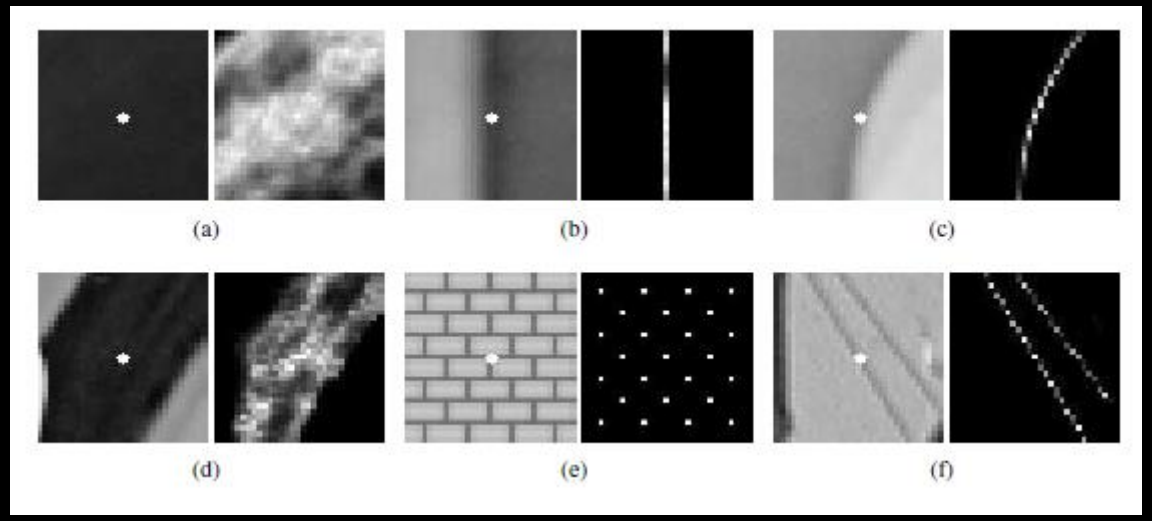
$$f(p,q) = e^{-\frac{|B(q)-B(p)|^2}{h^2}}$$

$$NL[u](x) = \frac{1}{C(x)} \int_\Omega e^{-\frac{(G_a * |u(x+.)-u(y+.)|^2)(0)}{h^2}}\, u(y)\, dy,$$

$$x \in \Omega,\ C(x) = \int_\Omega e^{-\frac{(G_a * |u(x+.)-u(z+.)|^2)(0)}{h^2}}\, dz$$

NL algorithm





BM3D

Dabov, K., Foi, A., Katkovnik, V., & Egiazarian, K. (2007). Image denoising by sparse 3-D transform-domain collaborative filtering. *IEEE Transactions on image processing*, *16*(8), 2080-2095.

Lebrun, M. (2012). An analysis and implementation of the BM3D image denoising method. *Image Processing On Line*, *2012*, 175-213.

NL algorithm

A. Buades, B. Coll and J. . -M. Morel, "A non-local algorithm for image denoising," *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, San Diego, CA, USA, 2005, pp. 60-65 vol. 2, doi: 10.1109/CVPR.2005.38.