Neighborhood filters, PDE's and the NL-means algorithm.

Toni Buades, Tomeu Coll, Universitat Illes Balears, JMM, ENS Cachan.

- Explore the idea of denoising by averaging groups of "similar" pixels
- Links denoising to grouping in the gestalt sense
- Understand (and use) the link with P.D.E. models, in particular the Perona-Malik equation

Figure 1: "Adding noise". Attneave, 1956: first white noise image drawn by hand, proposition that our perception only retains average and variance in white noise. No structure in noise

Denoising algorithms

- A denoising method D_h is a decomposition

$$v = D_h v + n(D_h, v), \tag{1}$$

where

  - h is a filtering parameter which usually depends on the standard deviation of the noise σ.
  - $D_h v$ is more smooth than the observed image v.
  - $n(D_h, v)$ is the noise guessed by the method.

Ideally, $n(D_h, v)$ looks like "a white noise". In particular, does not contain what we can suspect to be information in the image.

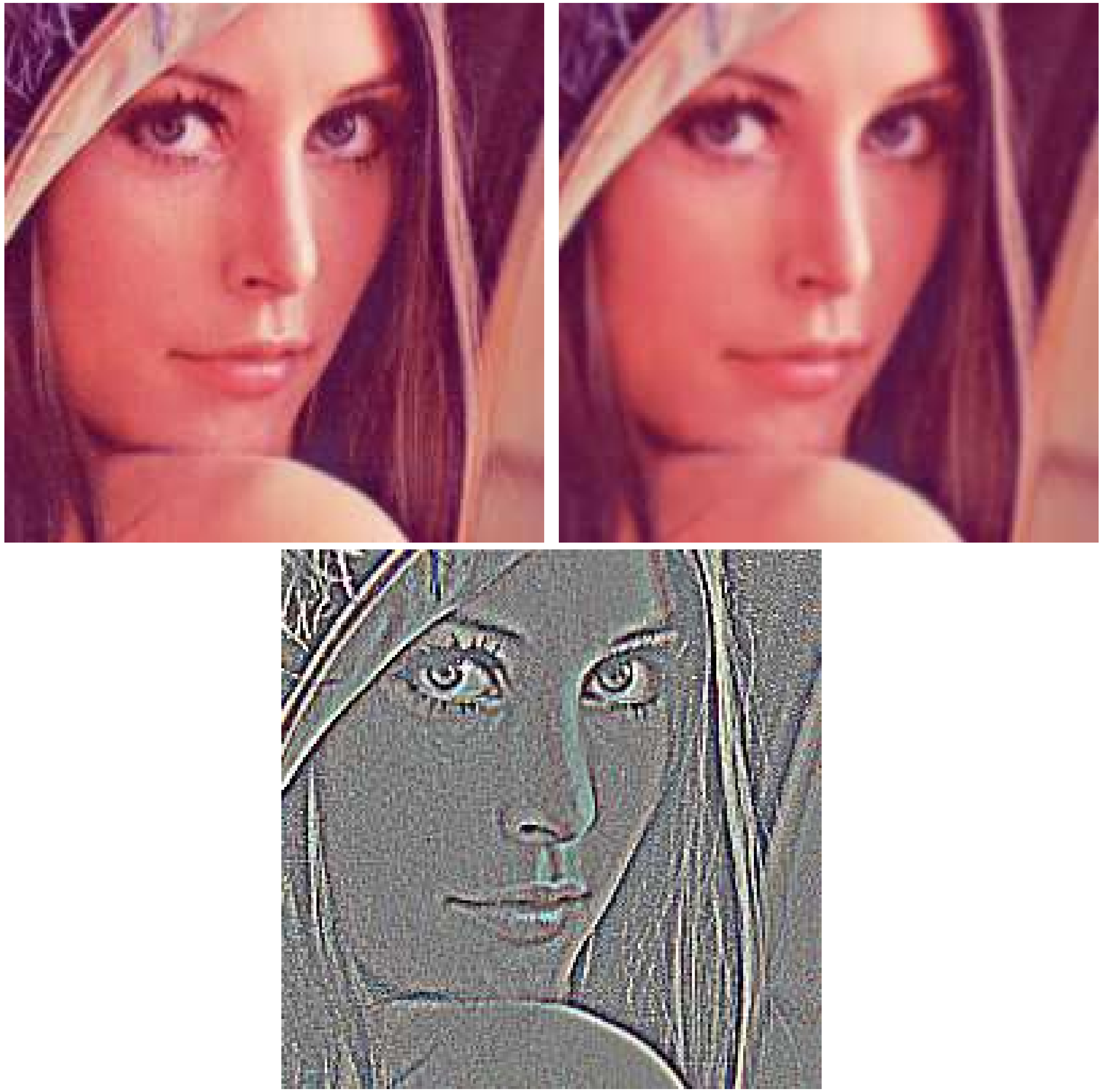


Figure 2: Lena (1973), A decomposition example : gaussian filtering

Image Filters: Gaussian filtering

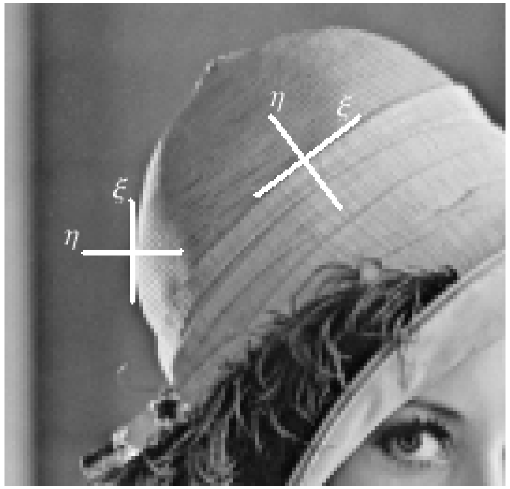
- If X_i are i.i.d of standard deviation σ

$$\mathrm{Var}\ \left(\frac{X_1 + \cdots + X_m}{m}\right) = \frac{\sigma^2}{m}$$

  Noise is reduced by averaging.

Image Filters II: Directional Filtering (Perona, Malik, Rudin, Osher, Fatemi, Weickert, Alvarez, Lions,....

- Local frame : the vectors $\eta = \frac{Du}{|Du|}$ and $\xi = \frac{Du^{\perp}}{|Du|}$ are respectively orthogonal and parallel to the level line passing trough x.

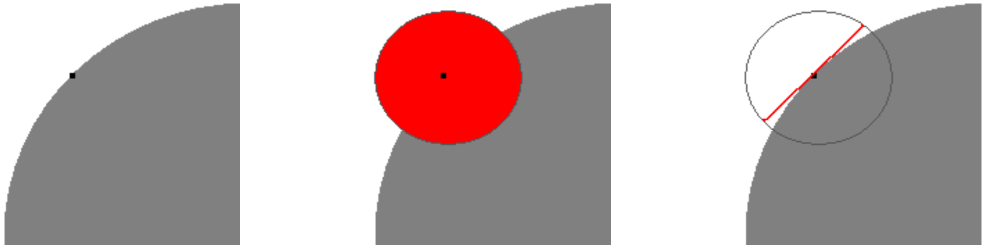


- Average of spatially close pixels in the direction of the level line.

$$AF_h u(\mathbf{x}) = G_h * u_{|l(\vec{\xi})} = \int_{\mathbb{R}} G_h(t) u(\mathbf{x} + t\vec{\xi}) dt,$$

where G_h is the one-dimensional Gauss function of standard deviation h.

- Difference with gaussian filter:



- The straight edges are well restored while flat and textured regions are degraded.

Image Filters III: Yaroslavski Neighborhood filter (sigma, bilateral).

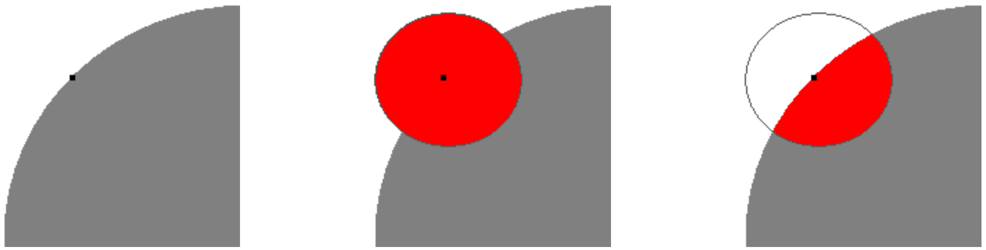
- Average of pixels which are close spatially and for grey level.

$$YNF_{h,\rho}u(\mathbf{x}) = \frac{1}{C(\mathbf{x})} \int_{B_\rho(\mathbf{x})} e^{-\frac{|u(\mathbf{y})-u(\mathbf{x})|^2}{h^2}} u(\mathbf{y})d\mathbf{y},$$

  where $C(\mathbf{x})$ is a normalizing factor, $B_\rho(\mathbf{x})$ is a ball of center $\mathbf{x}$ and radius ρ and h is the filtering parameter.

- Gaussian filtering does not take into account grey level values.

$$G_h u(\mathbf{x}) = \frac{1}{C(\mathbf{x})} \int e^{-\frac{|\mathbf{y}-\mathbf{x}|^2}{h^2}} u(\mathbf{y})d\mathbf{y}$$



- Noise is reduced in flat zones but not near the discontinuities. Isolated noisy points are kept.

Figure 3: sigma-filter, denoised, method noise

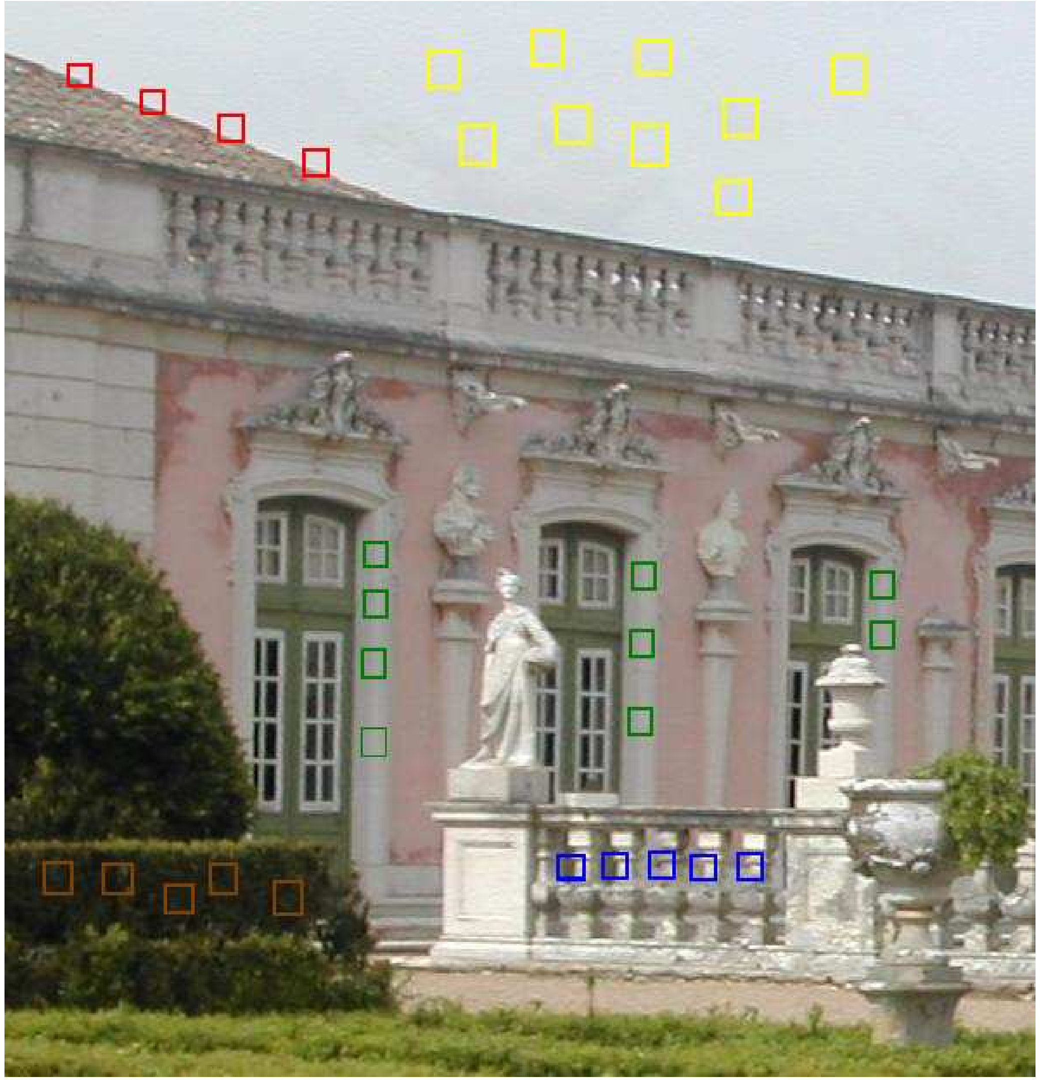


Figure 4: Groups of similar windows in a digital image, long range interaction. First used by Efros and Leung at UCB for texture synthesis

Figure 5: Pixels with a similar neighborhood will have a large weight (q1,q2) while pixels with a much different neighborhood have a small weight (q3).

The NL-means algorithm.

- NL-means filter. Average of pixels with a similar configuration in a whole gaussian neighborhood.

$$NL_h[u](\mathbf{x}) = \frac{1}{C(\mathbf{x})} \int_\Omega e^{-\frac{1}{h^2}\int_{\mathbb{R}^2} G_a(t)|u(\mathbf{x}+t)-u(\mathbf{y}+t)|^2 dt}\ u(\mathbf{y})\ d\mathbf{y},$$

  where G_a is a Gaussian kernel of standard deviation a and h acts as a filtering parameter.

- Non Local: Pixels of the whole image take part of the previous average.
- Neighborhood filters only compare the grey level value of neighboring pixels.

$$YNF_{h,\rho}u(\mathbf{x}) = \frac{1}{C(\mathbf{x})} \int_{B_\rho(\mathbf{x})} e^{-\frac{|u(\mathbf{y})-u(\mathbf{x})|^2}{h^2}} u(\mathbf{y}) d\mathbf{y},$$

Figure 6: NL-means denoising and "method noise"

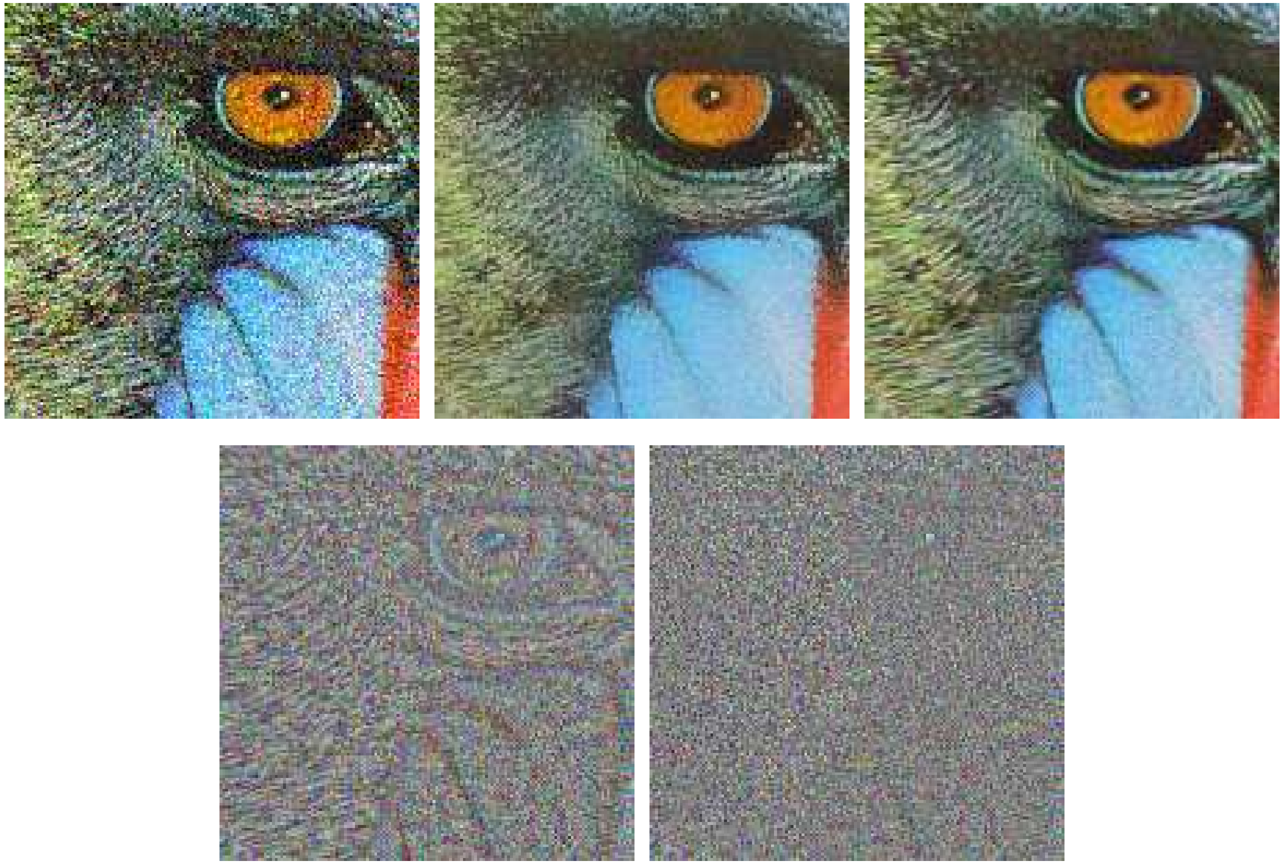
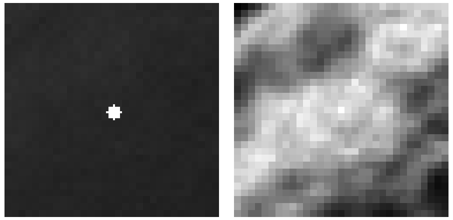


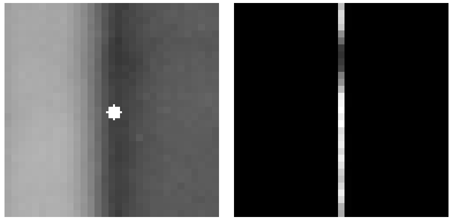
Figure 7: Sigma-filter, NL-means, method noise for each.

Probability distribution of similar pixels : gestalt groups!

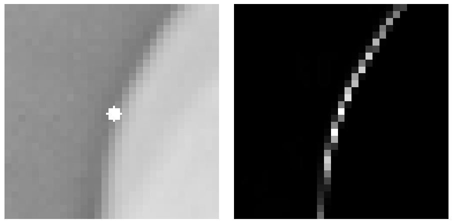
- Flat region. The large coefficients are spread out like in a convolution.



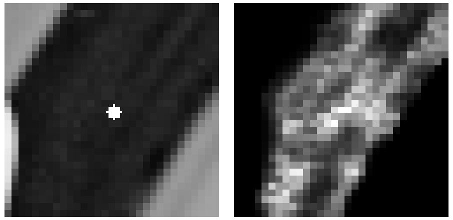
- Straight edge. The large coefficients are aligned like in a anisotropic filter.



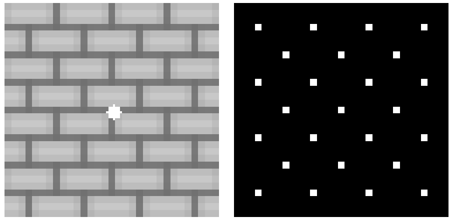
- Curved edge. The weights favor pixels belonging to the same contour.



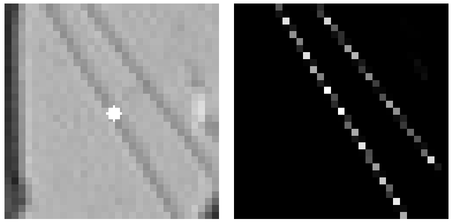
- Flat neighborhood. The average is made in the grey level neighborhood like the Yaroslavsky neighborhood filter.



- Periodic case. The large coefficients are distributed across the texture (non local).



- Repetitive structures. The weights favor similar configurations even they are far away (non local).



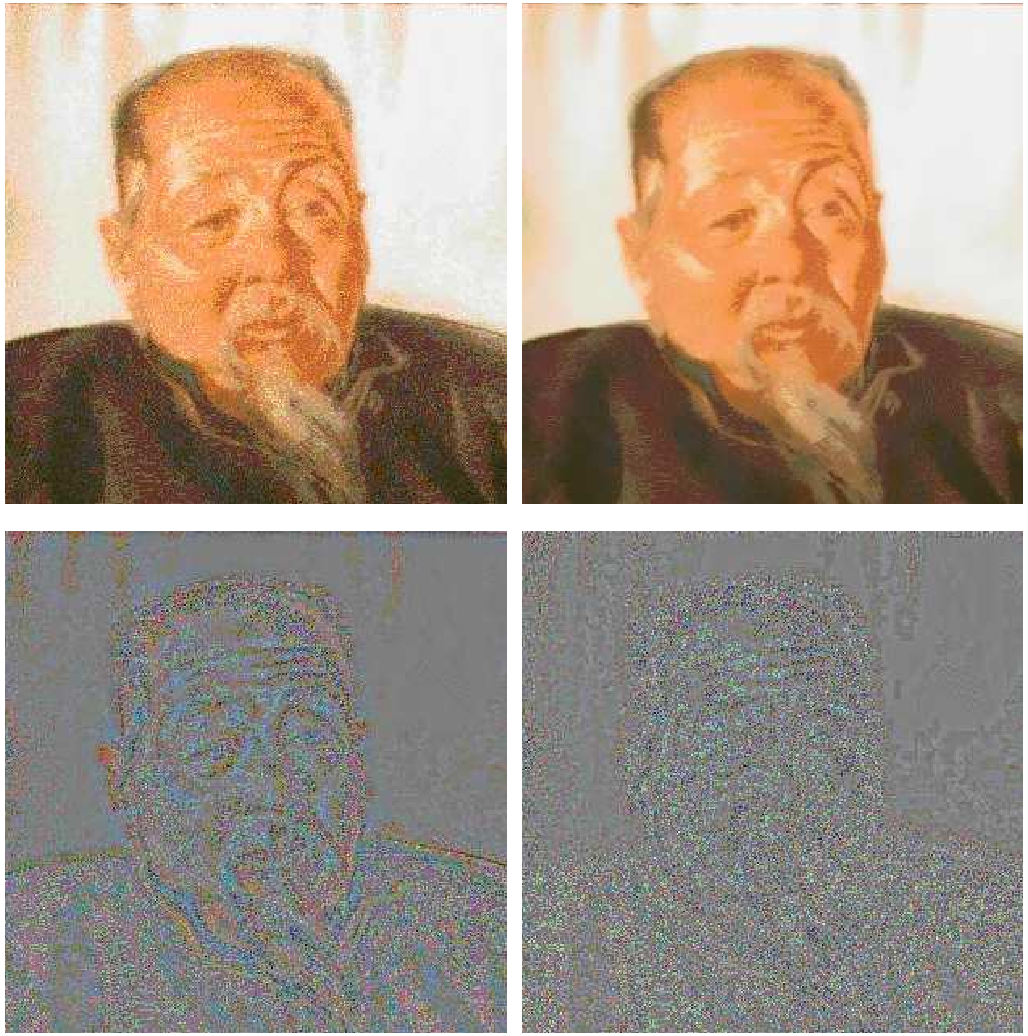
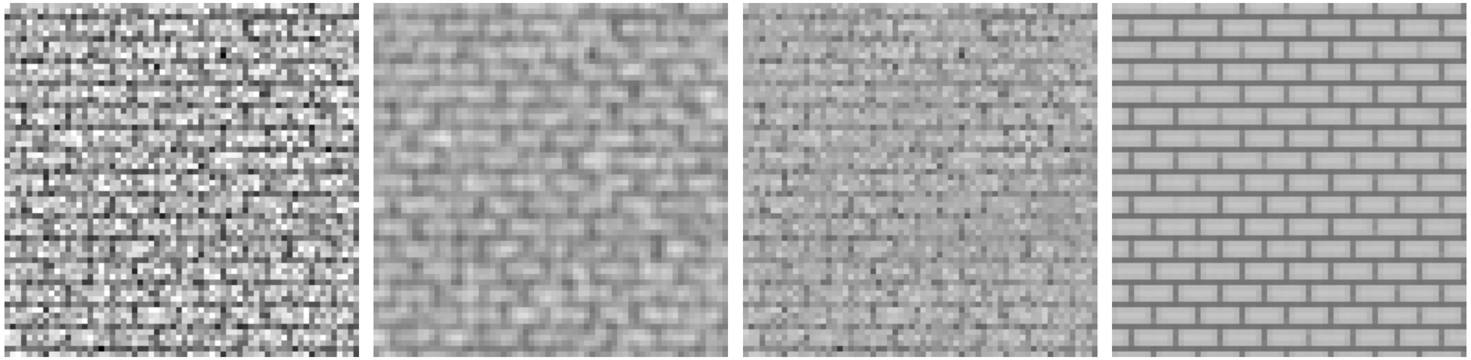


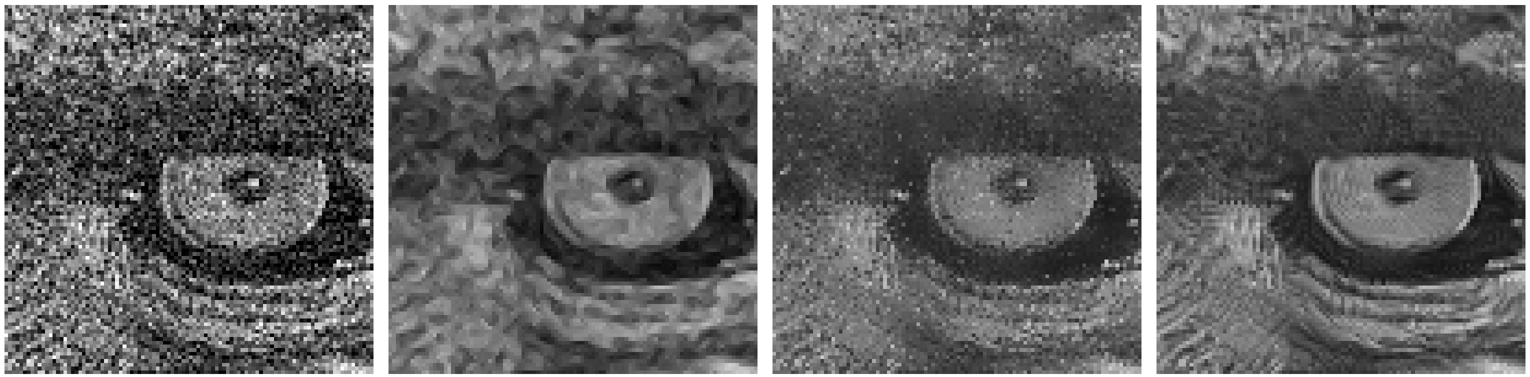
Figure 8: Real noisy image, signal dependent noise. Professional denoising software and NL-means.

Comparison: anisotropic, sigma filter, NL-means

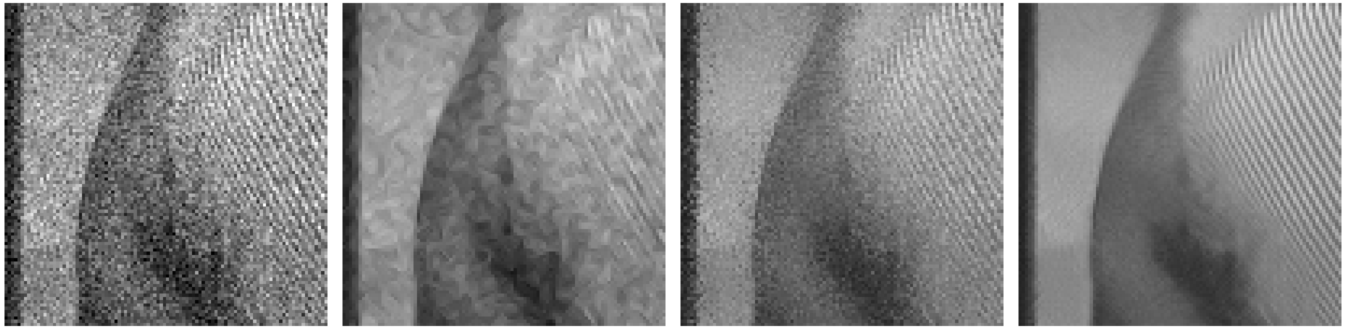
- Comparison on a periodic image



- Comparison on a natural texture.



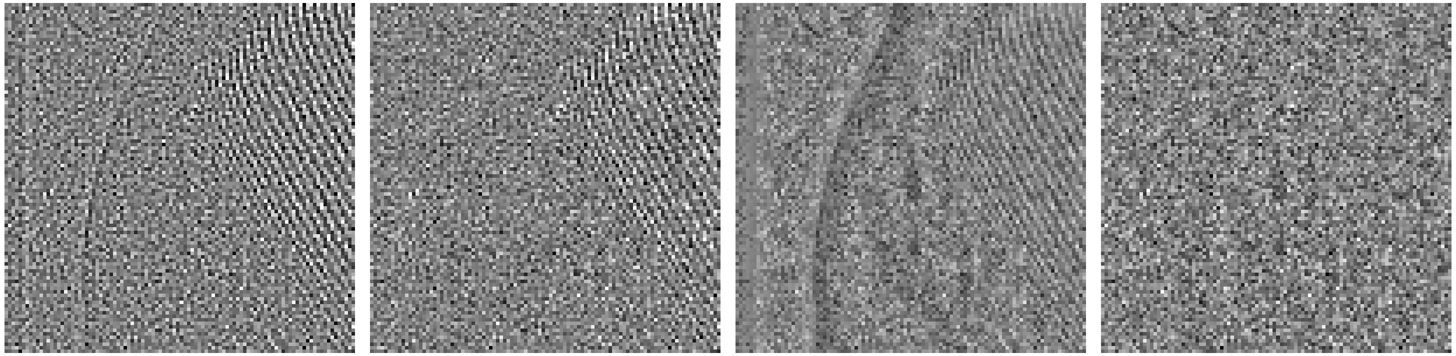
- Anisotropic, sigma-filter, NL-means.



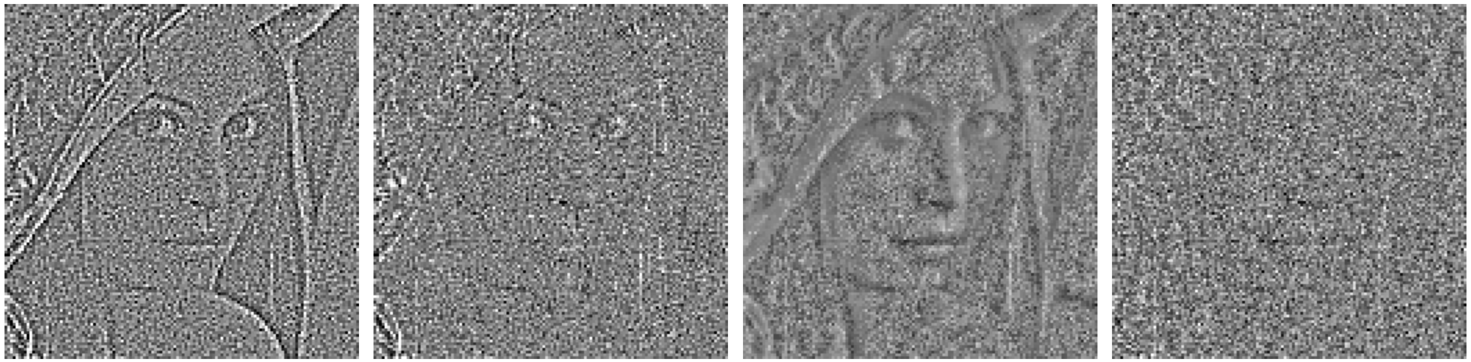
- Idem.

Comparison: Residual noise

- Residual noise on Barbara.



- Residual noise on Lena.



Problem: A shock effect

- The neighborhood filter and the NL-means share a shock effect.

  Flat zones and spurious boundaries are created.

- Objective: Analyze the enhancing behavior of the neighborhood filter and the NL-means algorithm.

PDEs filtering and enhancement.

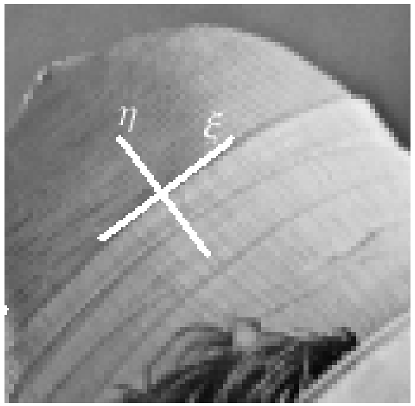
- The heat equation.

$$u_t = \Delta u.$$

The heat equation is an isotropic diffusion.

$$\Delta u = u_{\xi\xi} + u_{\eta\eta}$$

where $u_{\xi\xi}$ is the second derivative in the direction $\xi = Du^{\perp}/|Du|$ and $u_{\eta\eta}$ the second derivative in the the direction $\eta = Du/|Du|$.



$$u_{\eta\eta} = D^2 u(\frac{Du}{|Du|}, \frac{Du}{|Du|}), \quad u_{\xi\xi} = D^2 u(\frac{Du^{\perp}}{|Du|}, \frac{Du^{\perp}}{|Du|}),$$

Theorem 1 (Gabor 1960) *The convolution with a gaussian kernel G_h is such that*

$$G_h * u - u = h^2 \Delta u + o(h^2),$$

for h small enough.

- The mean curvature motion.

$$u_t = u_{\xi\xi}.$$

We diffuse only in the tangent direction and not across the edges.

Theorem 2 *The anisotropic filter AF_h is such that*

$$AF_h u(\boldsymbol{x}) - u(\boldsymbol{x}) = \frac{1}{2} u_{\xi\xi} h^2 + o(h^2),$$

where the relation holds when $Du(\boldsymbol{x}) \neq 0$.

- The inverse heat equation. Reversing the time we deblur the image.

$$u_t = -\Delta u = -u_{\xi\xi} - u_{\eta\eta}$$

It is ill posed.

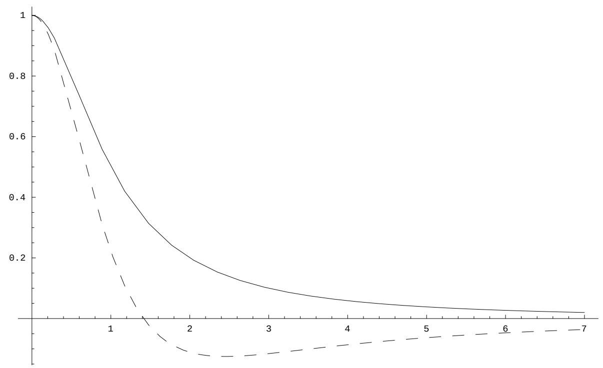
- The Perona-Malik equation: a filtering/enhancing equation.

$$u_t = \operatorname{div}(g(|Du|^2)Du),$$

where $g : [0, +\infty) \to [0, +\infty)$ is a smooth decreasing function satisfying $g(0) = 1$, $\lim_{s \to +\infty} g(s) = 0$.

The equation rewrites as

$$u_t = g(|Du|^2)u_{\xi\xi} + h(|Du|^2)u_{\eta\eta}.$$



g ——— h - - -

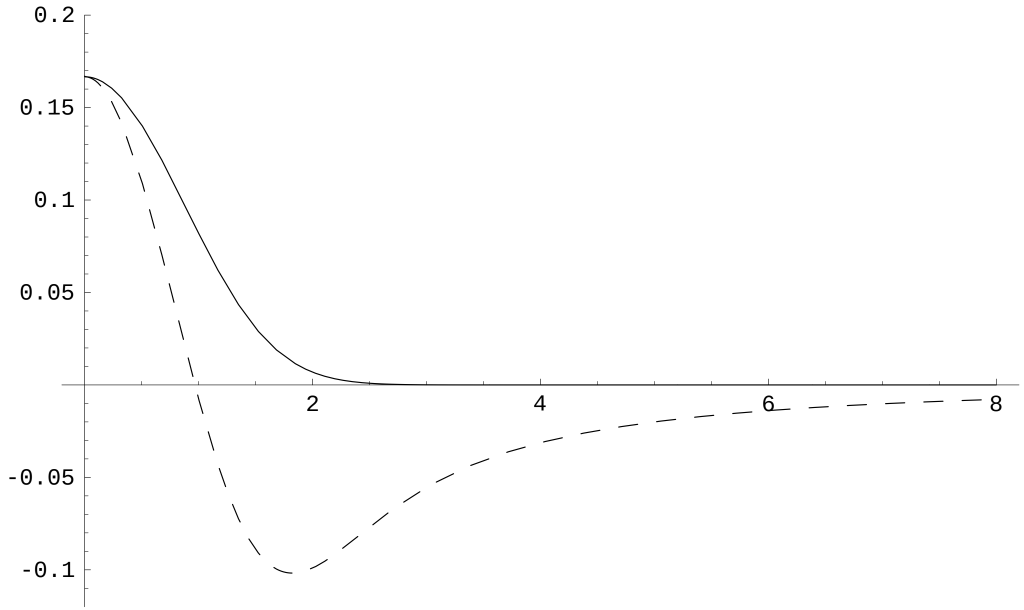
- One dimensional diffusion term in the orthogonal direction to the gradient.

- One dimensional diffusion in the gradient direction when $|Du|^2 < k$ and a reverse heat equation term when $|Du|^2 > k$.

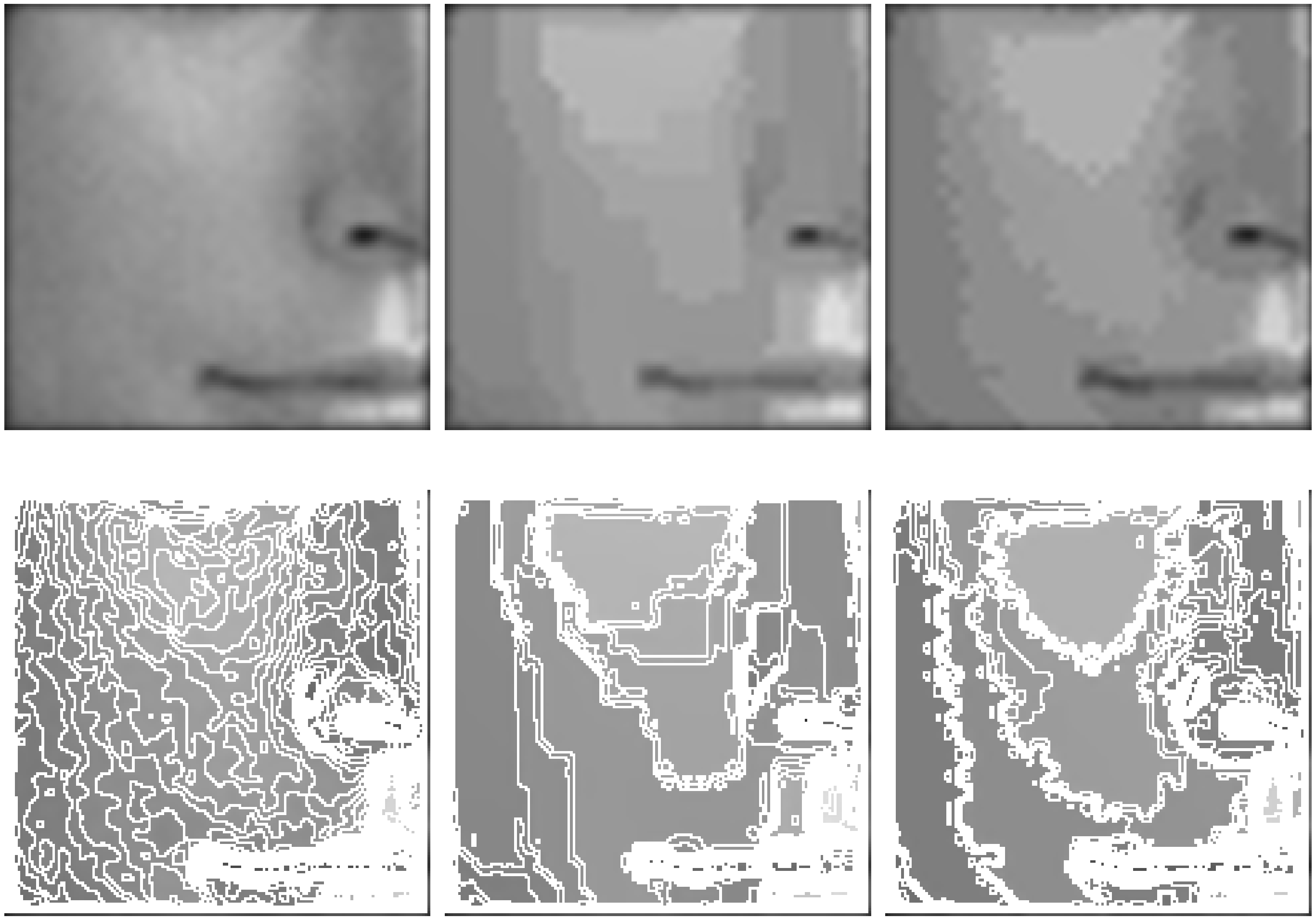
Neighborhood filters and PDEs: 2D

Theorem 3 *Suppose* $u \in C^2(\Omega)$, *and let* ρ, h, *such that* $\rho, h \to 0$ *and* $h = O(\rho)$. *Then, for* $x \in \Omega$,

$$YNF_{h,\rho}u(\boldsymbol{x}) - u(\boldsymbol{x}) \simeq \left[\tilde{g}(\frac{\rho}{h}\,|Du(\boldsymbol{x})|)\; u_{\xi\xi}(\boldsymbol{x}) + \tilde{f}(\frac{\rho}{h}\,|Du(\boldsymbol{x})|)\; u_{\eta\eta}(\boldsymbol{x})\right]\rho^2$$

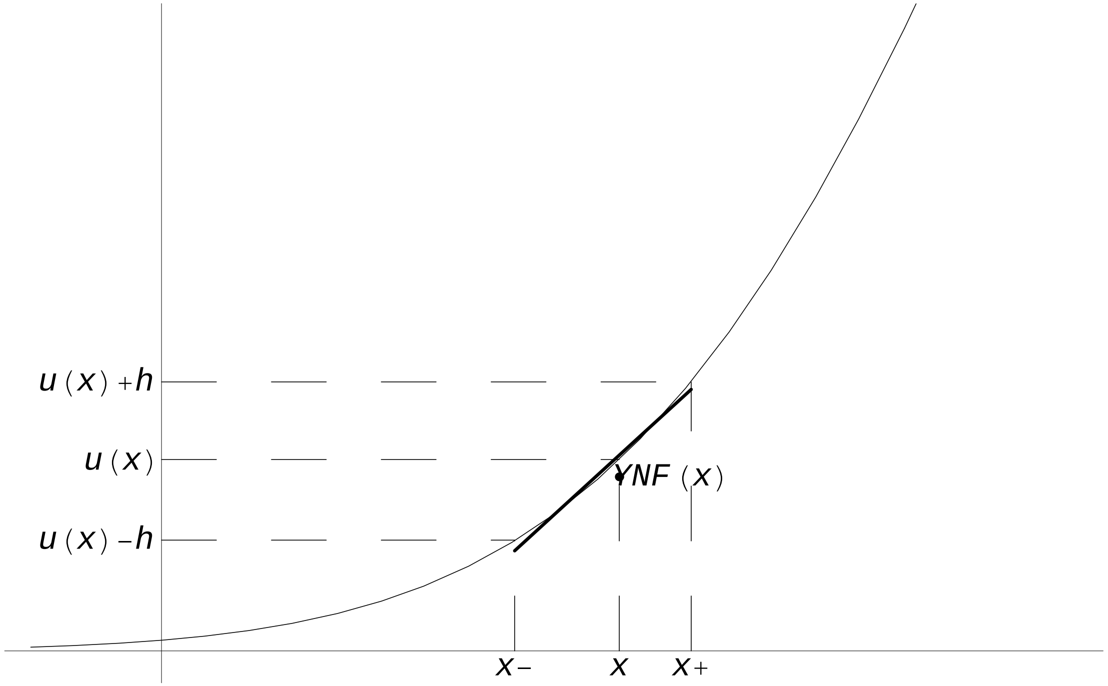


- Diffusion in the tangent direction with decreasing magnitude.
- Filtering/enhancing algorithm in the normal direction depending on $|Du|$. If $\tilde{B}$ denotes the zero of $\tilde{f}$, then a filtering model is applied wherever $|Du| < \tilde{B}\frac{h}{\rho}$ and an enhancing strategy wherever $|Du| > \tilde{B}\frac{h}{\rho}$.
- The intensity of the filtering in the tangent diffusion and the enhancing in the normal diffusion tend to zero when the gradient tends to infinity. Thus, points with a very large gradient are not altered.



The level lines of the Perona-Malik filter and the neighborhood filter tend to concentrate, creating flat zones. (Same effect for total variation denoising).

Geometrical explanation



The number of points y satisfying $u(x) - h < u(y) \leq u(x)$ is larger than the number satisfying $u(x) \leq u(y) < u(x) + h$. Thus, the average value $YNF(x)$ is smaller than $u(x)$, enhancing that part of the signal. The regression line of u inside (x_-, x_+) better approximates the signal at x.

Linear regression correction

- Locally approximate the image by a plane.
- The filtered value at $\mathbf{x} = (x_1, x_2)$ is given by $ax_1 + bx_2 + c$, where a, b, c minimize

$$\min_{a,b,c} \int_{B_\rho(\mathbf{x})} w(\mathbf{x}, \mathbf{y})(u(\mathbf{y}) - ay_1 - by_2 - c)^2 d\mathbf{y}$$

  and

$$w(\mathbf{x}, \mathbf{y}) = e^{-\frac{|u(\mathbf{y}) - u(\mathbf{x})|^2}{h^2}}.$$

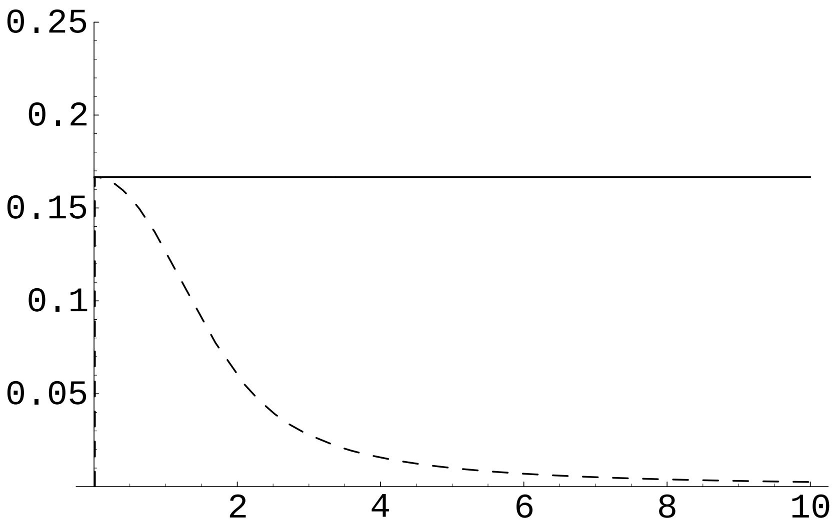
- Points with a grey level value close to $u(x)$ will have a larger influence in the minimization process than those with a further grey level value.

- **Theorem 4** *Suppose* $u \in C^2(\Omega)$, *and let* $\rho, h > 0$ *such that* $\rho, h \to 0$ *and* $O(\rho) = O(h)$. *Let* $\tilde{g}$ *be the continuous function defined as* $\tilde{g}(0) = \frac{1}{6}$,

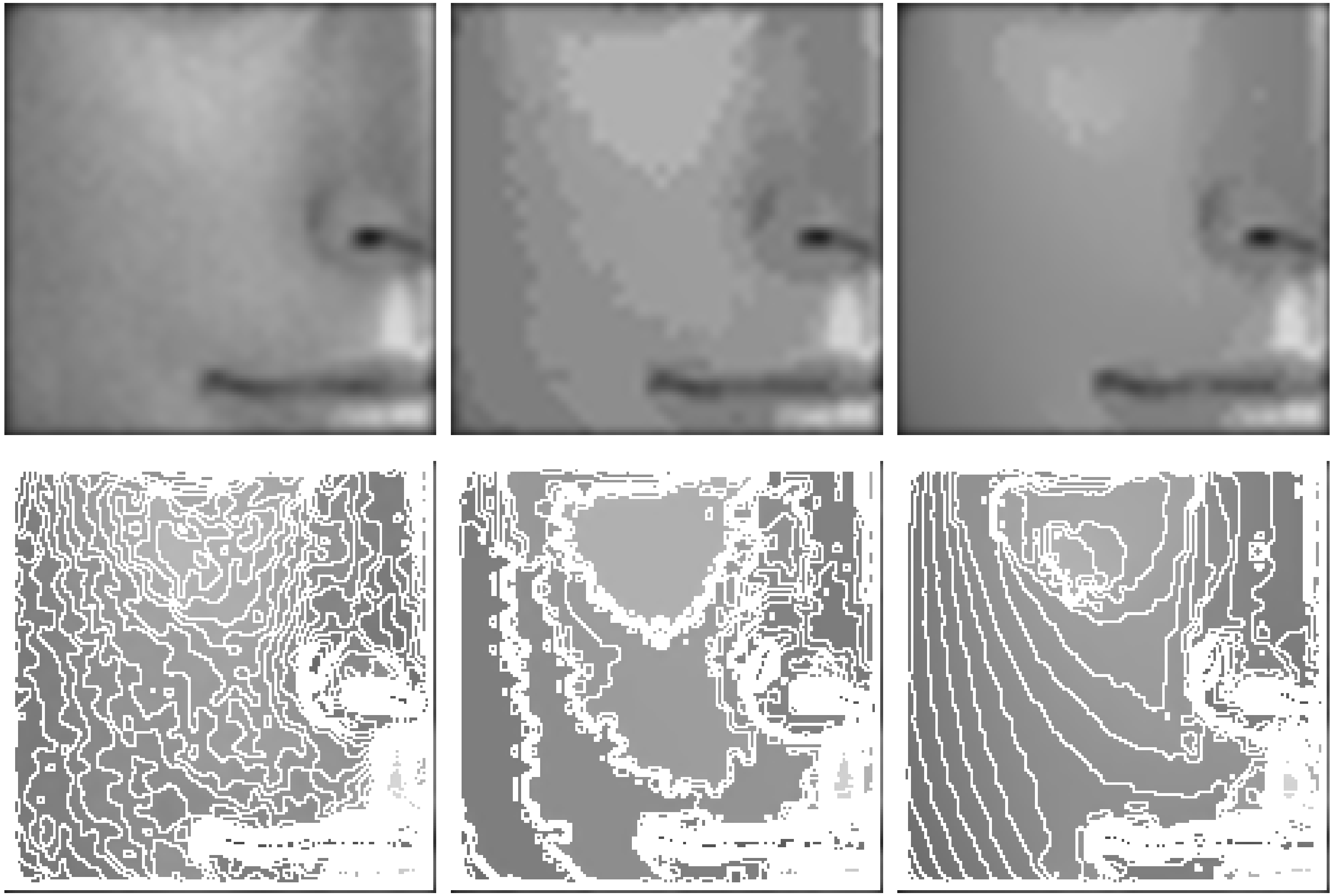
$$\tilde{g}(t) = \frac{8t^2 e^{-2t^2} - 8te^{-t^2} E(t) + 2E(t)^2}{t^2(4E(t)^2 - 8te^{-t^2} E(t))}$$

for $t \neq 0$, *where* $E(t) = 2\int_0^t e^{-s^2}\, ds$. *Then*

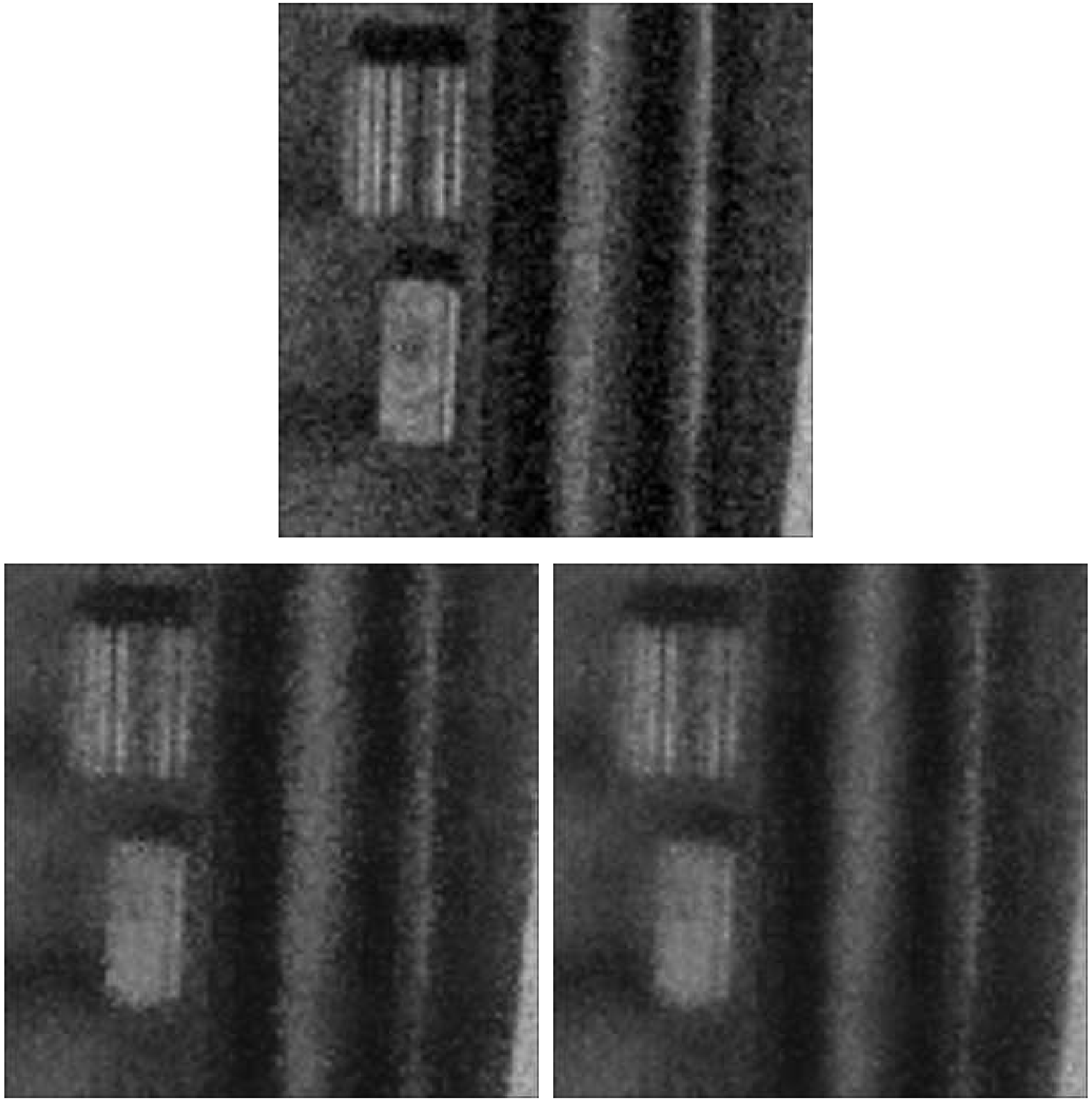
$$LYNF_{h,\rho} u(\boldsymbol{x}) - u(\boldsymbol{x}) \simeq (\frac{1}{6} u_{\xi\xi} + \tilde{g}(\frac{\rho}{h}\,|Du|) u_{\eta\eta})\,\rho^2.$$



- The tangent diffusion is a positive constant and the normal diffusion is positive and decreasing, and therefore there is no enhancing effect.
- The algorithm combines the tangent and normal diffusions wherever the gradient is small. Wherever the gradient is larger the normal diffusion is canceled and the image is filtered only in its tangent direction.
- In the Perona-Malik model the diffusion is stopped near the edges. In this case, the edges are filtered by a mean curvature motion.



The regression correction filters the level lines by a curvature motion



The linear correction doesn't create shocks contours.

The Nl-means linear regression correction

- In order to apply the regression correction to the NL-means algorithm we restrict the search zone for a pixel $\mathbf{x} = (x_1, x_2)$ to a neighborhood $B_\rho(\mathbf{x})$.
- The filtered value is given by $ax_1 + bx_2 + c$, where a, b, c minimize

$$\min_{a,b,c} \int_{B_\rho(\mathbf{x})} w(\mathbf{x},\mathbf{y})(u(\mathbf{y}) - ay_1 - by_2 - c)^2 d\mathbf{y}$$

and

$$w(\mathbf{x},\mathbf{y}) = e^{-\frac{1}{h^2}\int_{\mathbb{R}^2} G_a(t)|u(\mathbf{x}+t)-u(\mathbf{y}+t)|^2 dt}.$$

- Neighborhood filter, without and with linear correction

- NL-means, without and with linear correction

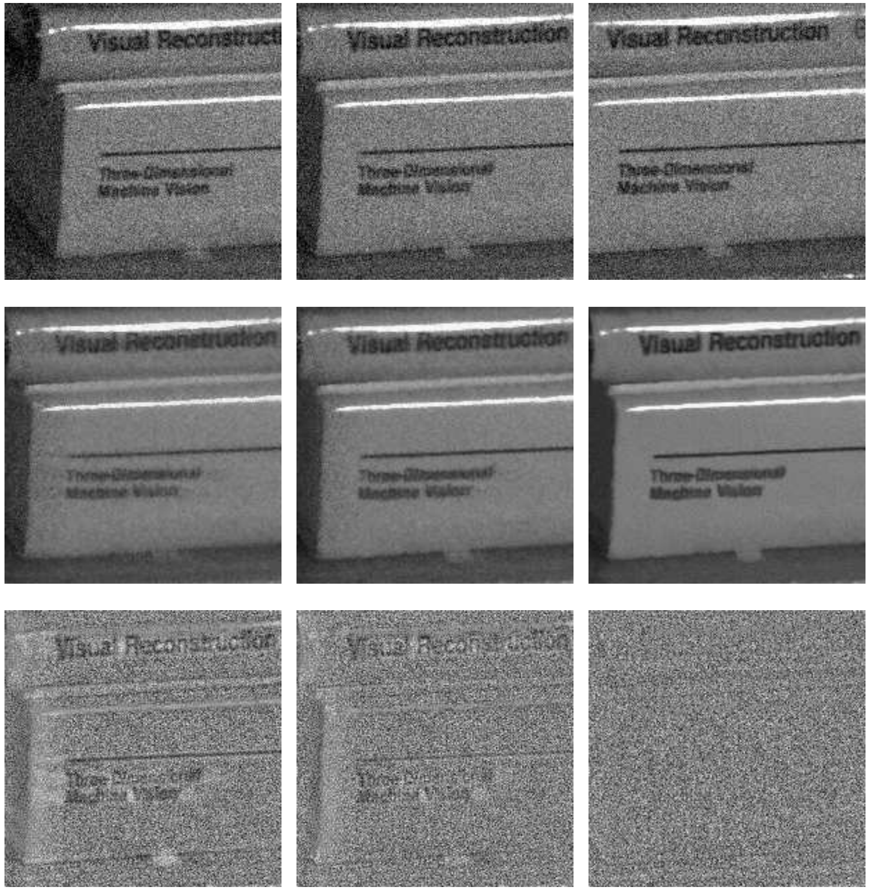


Figure 9: “Against motion compensation”. Three images of a movie + noise. Second row: denoising of the central frame by AWA, by NL-means on trajectory and by general NL-means. Bottom : method noise.

Preprints available at CMLA:

- N^o 2004-15 : On image denoising methods.
- N^o 2005-04 : Neighborhood filters and PDE's.

Consistency of the NL-means algorithm

- Hypothesis: As the size of the image grows we can find many similar samples for all the details of the image (stationarity assumptions).
  - Let Z denote the sequence of random variables $Z_i = \{V(i), V(\mathcal{N}_i \backslash \{i\})\}$
  - Let denote $\hat{NL}_n$ the NL-means algorithm applied to the subsequence of Z, $Z_n = \{V(i), V(\mathcal{N}_i \backslash \{i\})\}_{i=1}^{n}$ and where $v(\mathcal{N}_i \backslash \{i\})$ is used to compute the weights instead of $v(\mathcal{N}_i)$.
  - Let $r(i)$ denote $E[V(i) \mid V(\mathcal{N}_i \backslash \{i\}) = v(\mathcal{N}_i \backslash \{i\})]$.
- **Theorem 5 (Conditional expectation theorem)** *Let* $Z = \{V(i), V(\mathcal{N}_i \backslash \{i\})\}$ *for* $i = 1, 2, \ldots$ *be a strictly stationary and mixing process. Then,*

$$|NL_n(i) - r(i)| \to 0 \quad a.s$$

for $i \in \{1, \ldots, n\}$.

- Additive white noise model

Theorem 6 *Let V, U, N be random fields on I such that $V = U + N$, where N is a signal independent white noise. Then, the following statements are hold.*

(i) $E[V(i) \mid V(\mathcal{N}_i \backslash \{i\}) = x] = E[U(i) \mid V(\mathcal{N}_i \backslash \{i\}) = x]$ *for all* $i \in I$ *and* $x \in \mathbb{R}^p$.

(ii) The expected random variable $E[U(i) \mid V(\mathcal{N}_i \backslash \{i\})]$ *is the function of* $V(\mathcal{N}_i \backslash \{i\})$ *that minimizes the mean square error*

$$\min_g E[U(i) - g(V(\mathcal{N}_i \backslash \{i\}))]^2 \qquad (1)$$

Theorem 7 *Suppose* $u \in C^2$, *and let* $\rho, h, \alpha > 0$ *such that* $\rho, h \to 0$ *and* $h = O(\rho^\alpha)$. *Consider the continuous function* $g(t) = \frac{te^{-t^2}}{E(t)}$, *for* $t \neq 0$, $g(0) = \frac{1}{2}$, *where* $E(t) = 2\int_0^t e^{-s^2}\, ds$. *Let* f *be the continuous function*

$$f(t) = \frac{g(t)}{t^2} + g(t) - \frac{1}{2t^2}, \quad f(0) = \frac{1}{6}.$$

Then, for $x \in \mathbb{R}$,

1. *If* $\alpha < 1$,

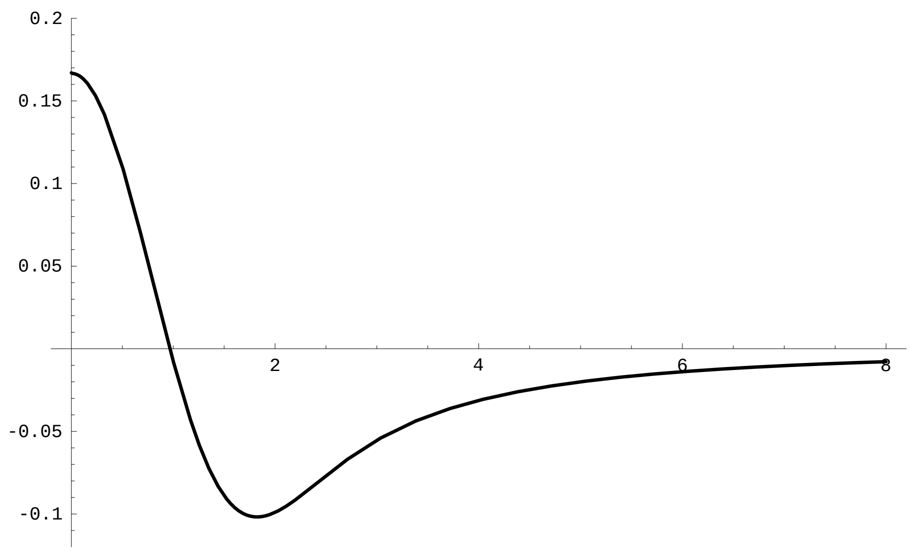
$$YNF_{h,\rho}u(x) - u(x) \simeq \frac{u''(x)}{6}\rho^2.$$

2. *If* $\alpha = 1$,

$$YNF_{h,\rho}u(x) - u(x) \simeq f(\frac{\rho}{h}\,|u'(x)|)\, u''(x)\, \rho^2.$$

3. *If* $1 < \alpha < \frac{3}{2}$,

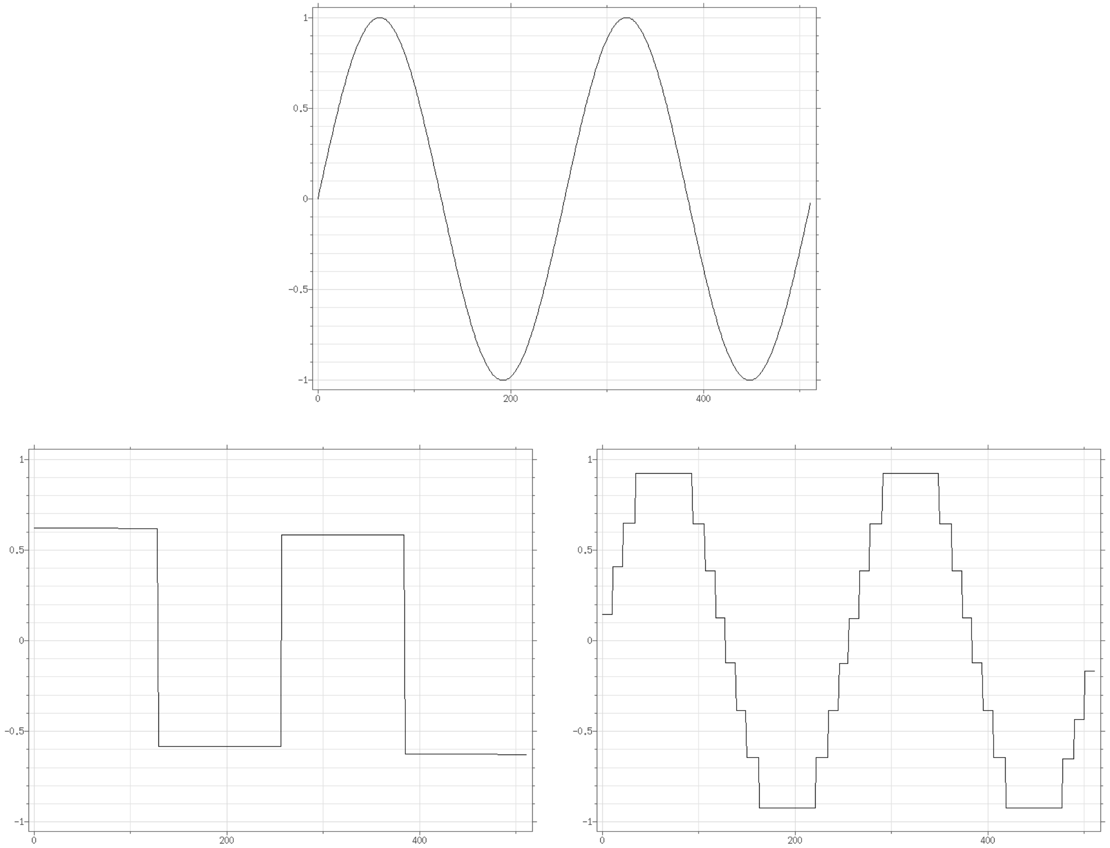
$$YNF_{h,\rho}u(x) - u(x) \simeq g(\rho^{1-\alpha}|u'(x)|)\, u''(x)\, \rho^2.$$



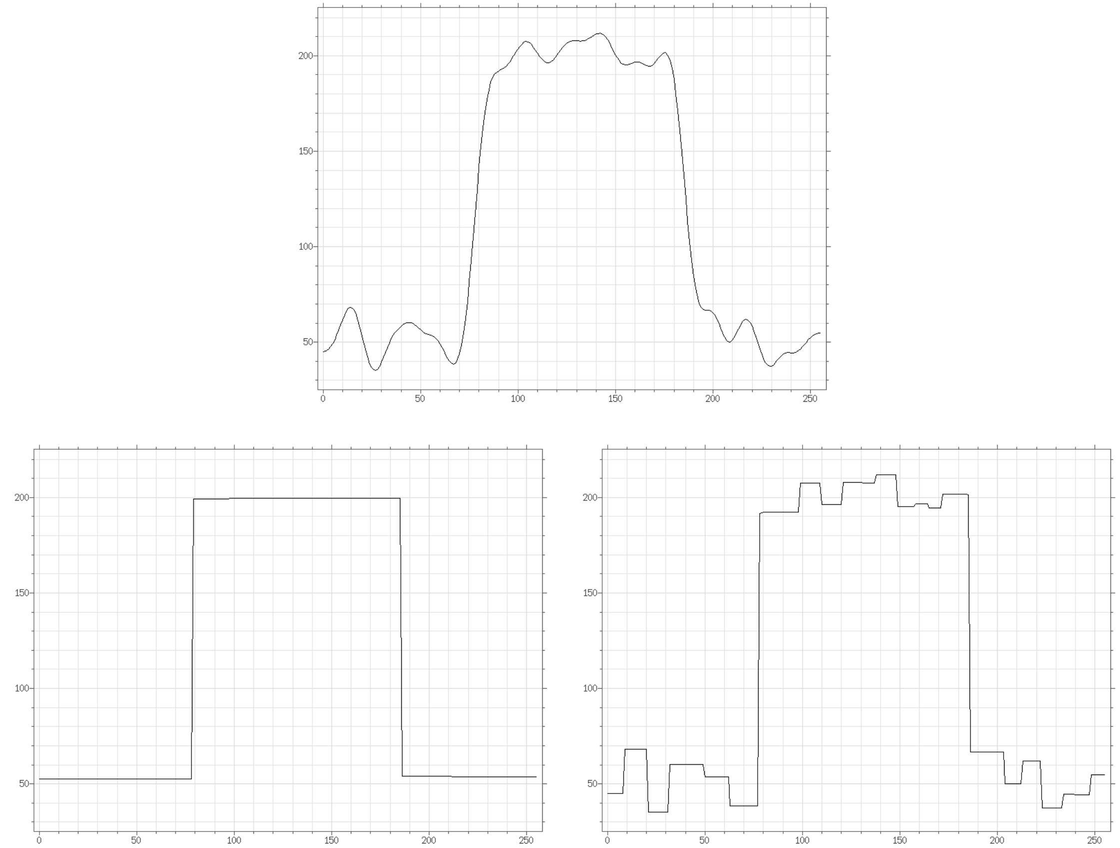
- The filter behaves as a filtering/enhancing algorithm depending on the magnitude of $|u'(x)|$.

- If B denotes the zero of f, then a filtering model is applied wherever $|u'| < B\frac{h}{\rho}$ and an enhancing model wherever $|u'| > B\frac{h}{\rho}$.

- Points x where $|u'(x)|$ is large are not altered.



Singularities are created due to the transition of smoothing to enhancement. The number of enhanced regions strongly depends upon the ratio $\frac{\rho}{h}$.



The shock filter is sensitive to the noise and creates spurious steps. The filtering/enhancing character of the neighborhood filter avoids this effect.