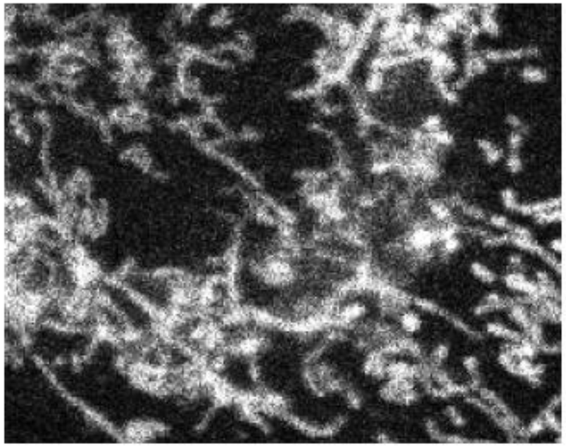
NL-means and patch-based methods

F. Tupin
Cours TADI

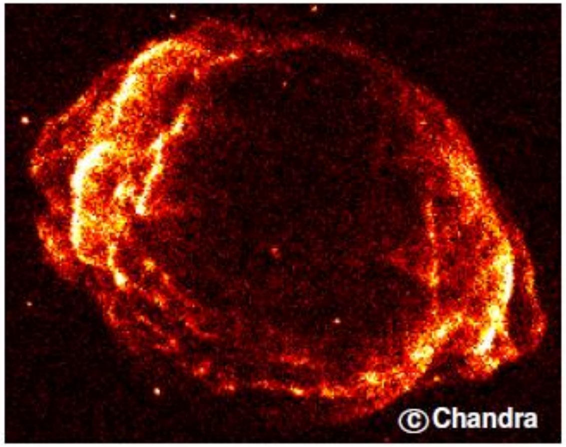
Acknowledgments: Y. Gousseau, C. Deledalle, V. Duval, C. Aguereberre, G. Tartavel

- **Introduction**
  - Denoising and models
- **Non-local / patch based approaches**
  - Principle
  - Toy examples
  - Limits and solutions
- **Advanced methods**
  - Noise adaptation
  - Iterative approaches
  - Automatic setting of parameters
  - Shape of patches

Image denoising



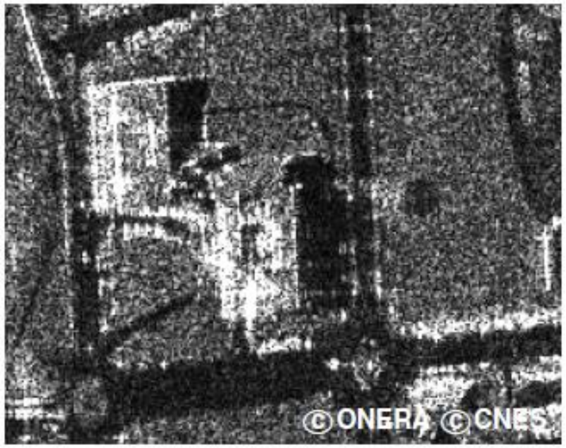
(a) Mitochondrion in microscopy



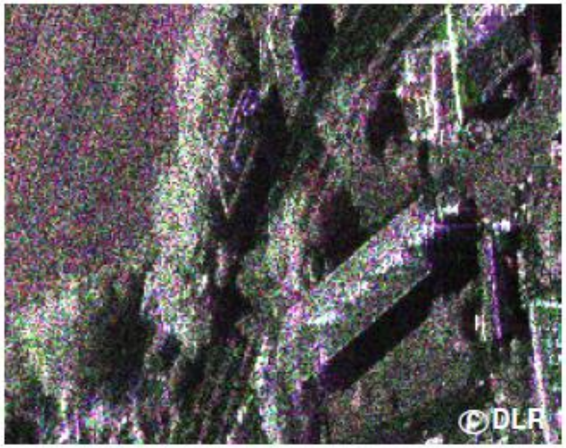
(b) Supernova in X-ray imagery

(c) Fetus using ultrasound imagery

(d) Plane wreckage in SONAR imagery

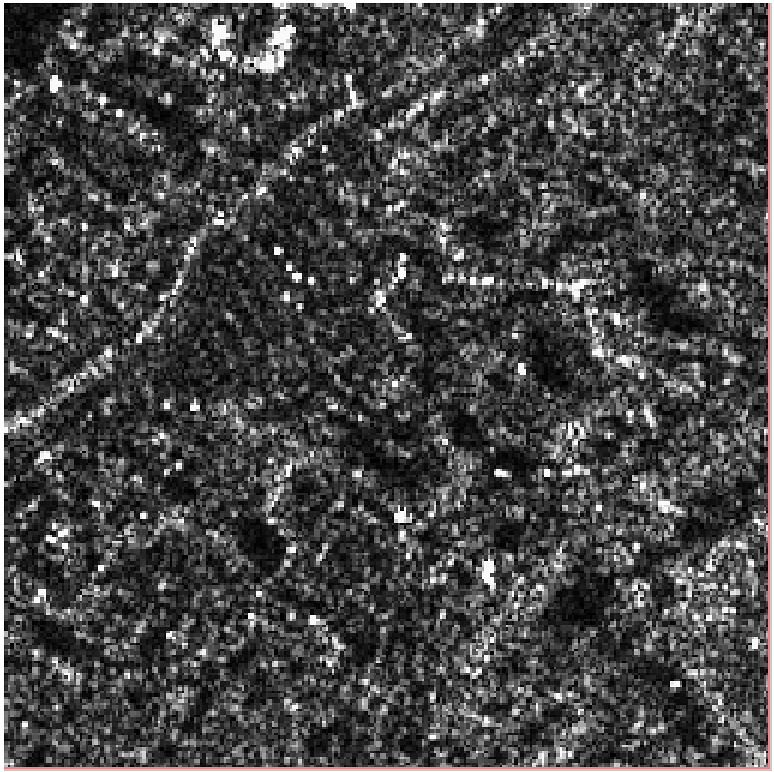
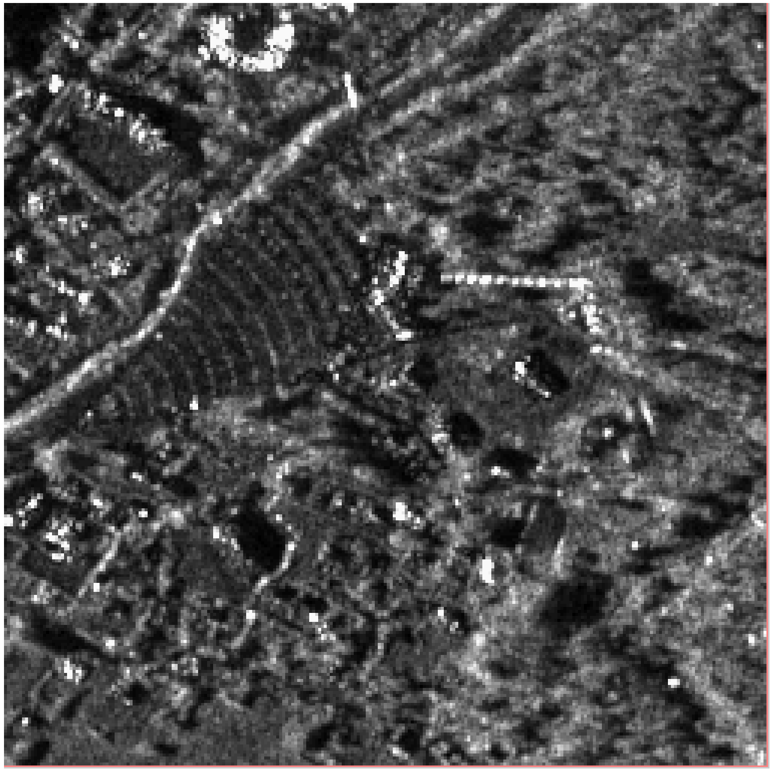


(e) Urban area using SAR imagery



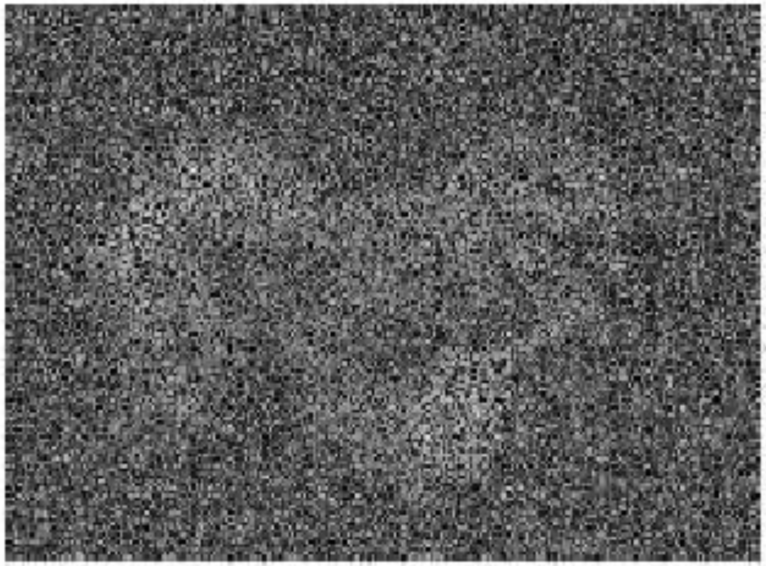
(f) Polarimetric SAR imagery

Can we denoise ?

Temporal information

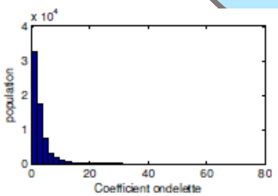
Can we denoise ?

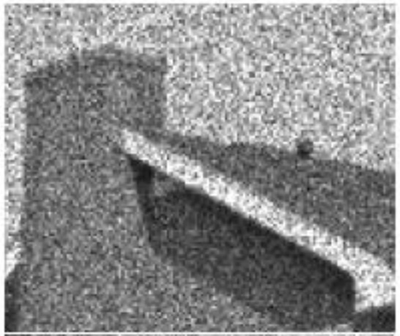


Spatial information

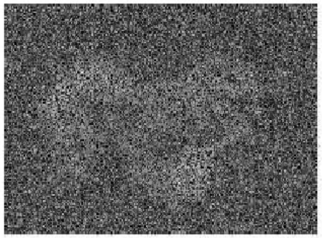
Image models

Hypothesis of signal / noise separation



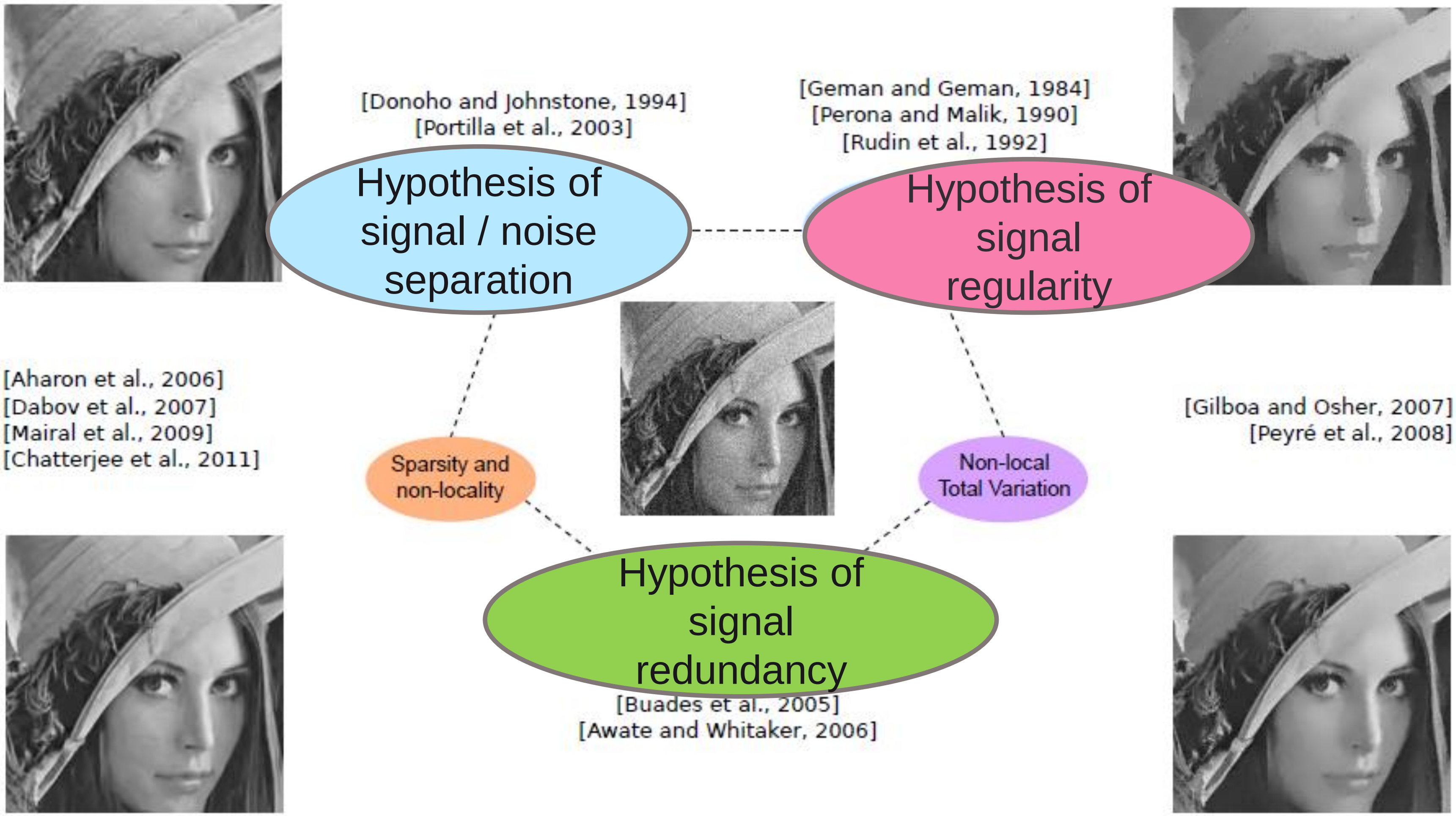


Hypothesis of signal smoothness



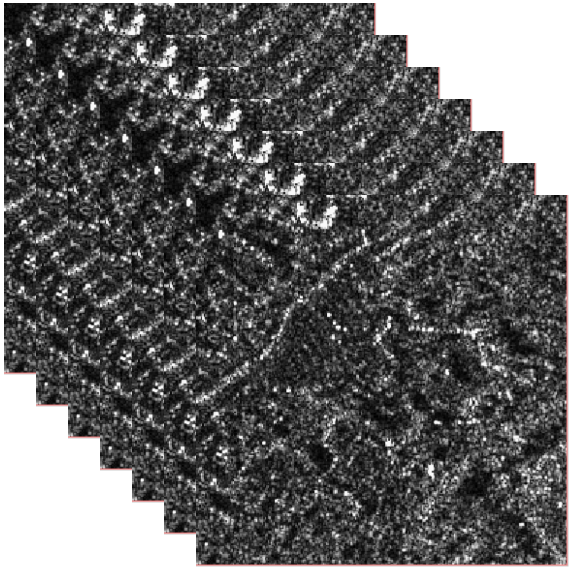
Hypothesis of signal redundancy

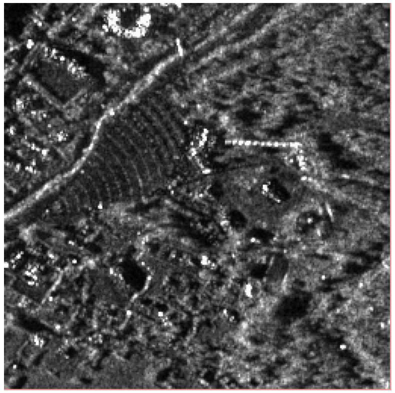
Image models



Patch-based approaches perform best (see review of [Katkovnik et al., 2010])

Denoising and « averaging »





- Average of many noisy values: estimation of the « true » reflectivity
- …only if the selected values are coming from the same underlying noise-free value…

➡ How can we select them on the image?

Selection based filtering

- **Where finding the « good » information?**

Locally (linear filtering) | Locally (anisotropic diff.) | Oracle

- **Introduction**
  - Denoising and models
- **Non-local / patch based approaches**
  - Principle
  - Toy examples
  - Limits and solutions
- **Advanced methods**
  - Noise adaptation
  - Iterative approaches
  - Automatic setting of parameters
  - Shape of patches

Selection-based filtering

- **Non-local approaches:**
  - Relaxing locality and connexity constraints for pixel selection: selection based on similarity

$$\hat{u}_s = \sum_{t \in \Omega} w(s,t) v_t$$

u_s searched noise-free value
$\hat{u}_s$ estimated noise-free value
v_s observed noisy value

Selection-based filtering

Non-local approaches:

- Relaxing locality and connexity constraints for pixel selection: selection based on similarity [Yaroslavsky, 85]

$$\hat{u}_s = \sum_{t \in \Omega} w(s,t) v_t \qquad w(s,t) = \exp\left(-\frac{d(v_s, v_t)}{h^2}\right)$$

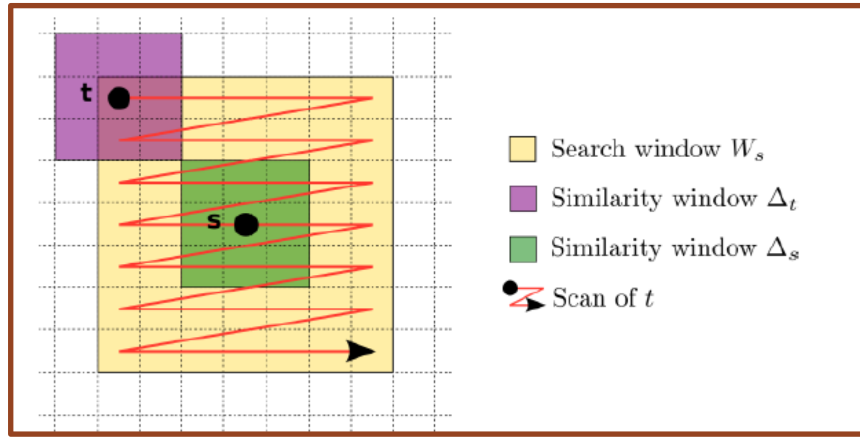
How computing d when having only noisy values ?

Use patches !

Non-local means [Buades 05]

- **Algorithm :**

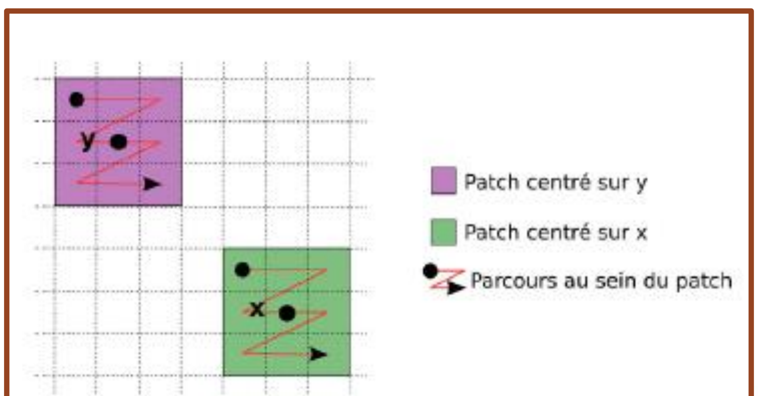
$$\hat{u}_s = \sum_{t\in\Omega} w(s,t)\, v_t$$



- Similarity of pixels = similarity of patches

$$w(x,y) = \frac{e^{-\frac{sim(x,y)}{2h^2}}}{\sum_z e^{-\frac{sim(x,z)}{2h^2}}}$$

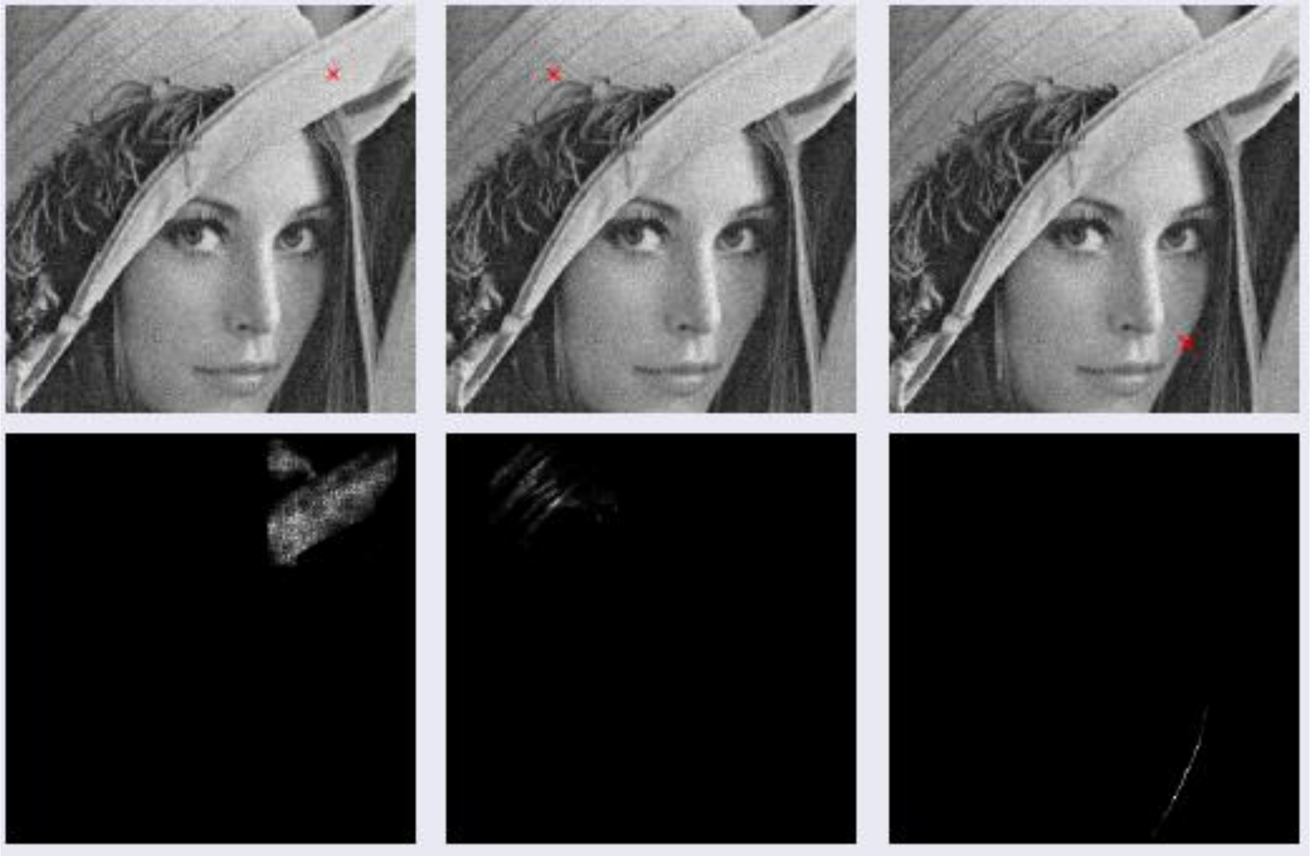
$$sim(x,y) = \frac{1}{s^2}\|\tilde{U}(x)-\tilde{U}(y)\|^2 \triangleq \frac{1}{s^2}\sum_{\delta}(\tilde{u}(x+\delta)-\tilde{u}(y+\delta))^2$$



$$w(x,y) = \frac{e^{-\frac{\|U(x)-U(y)\|^2}{2h^2}}}{\sum_{z\in W} e^{-\frac{\|U(x)-U(z)\|^2}{2h^2}}}$$

Selection-based filtering

■ **Non-local approaches: example of weight maps**



$$w(x,y) = \frac{e^{-\frac{\|U(x)-U(y)\|^2}{2h^2}}}{\sum_{z\in W} e^{-\frac{\|U(x)-U(z)\|^2}{2h^2}}}$$

Selection-based filtering

- **Non-local approaches: example of weight maps**

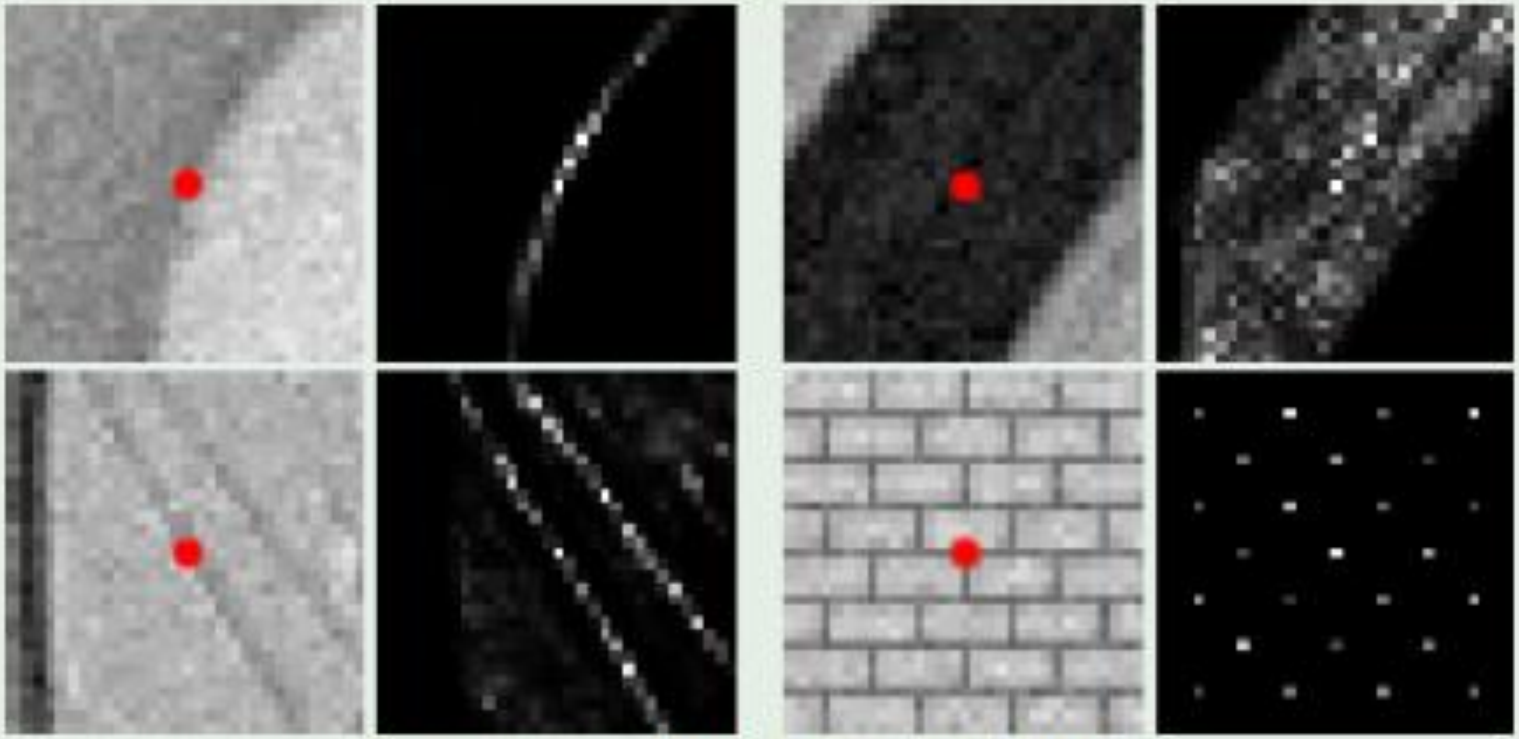
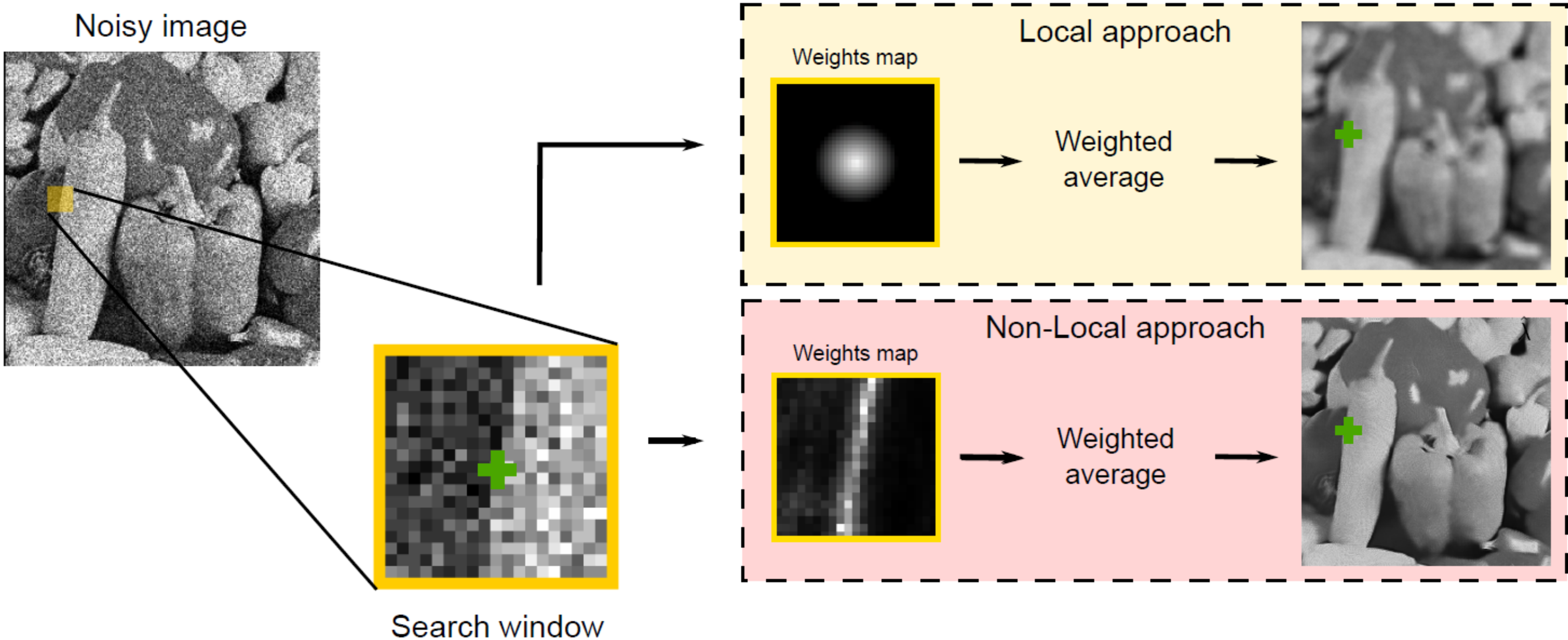
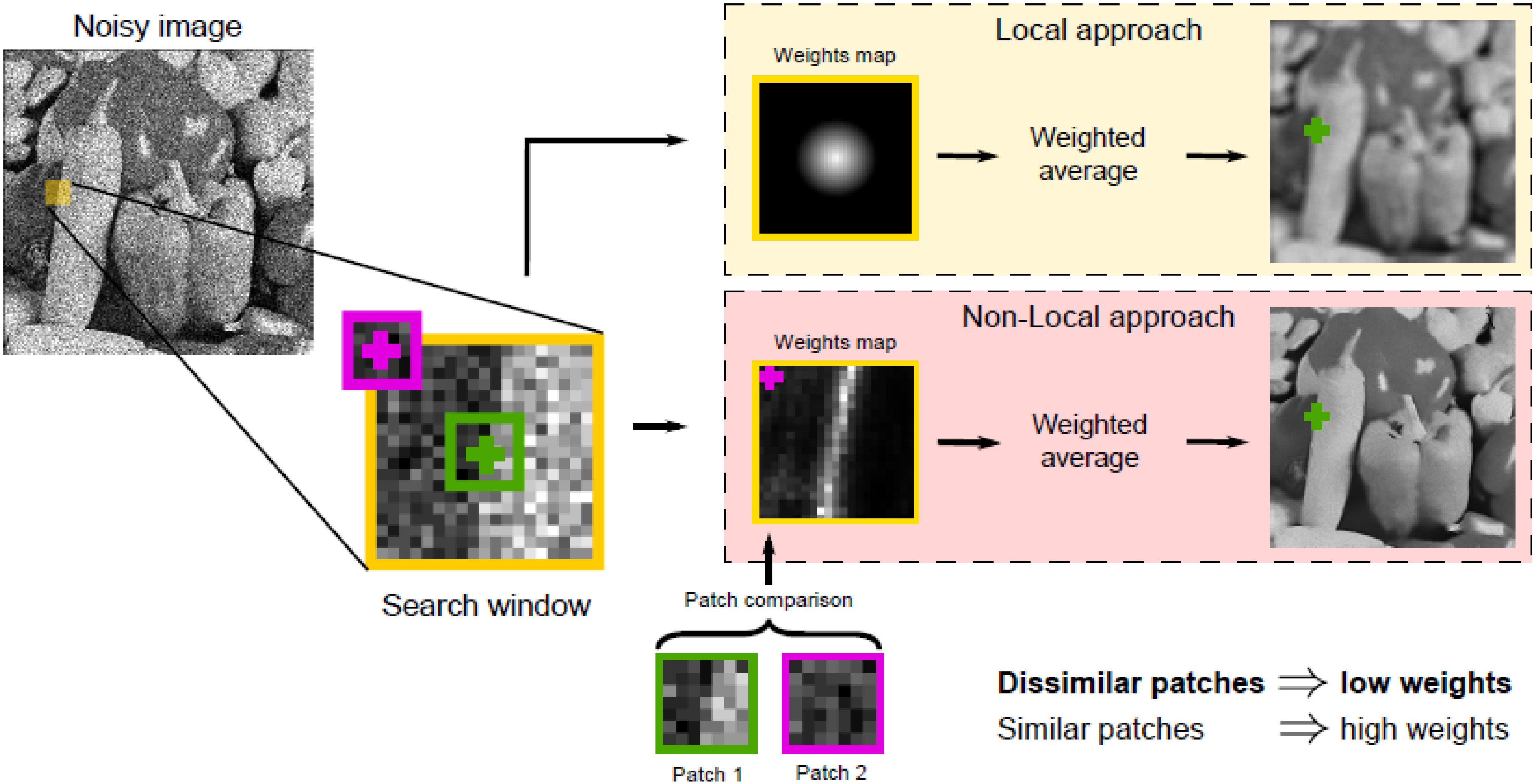


image extracted from [Buades et al., 2005]

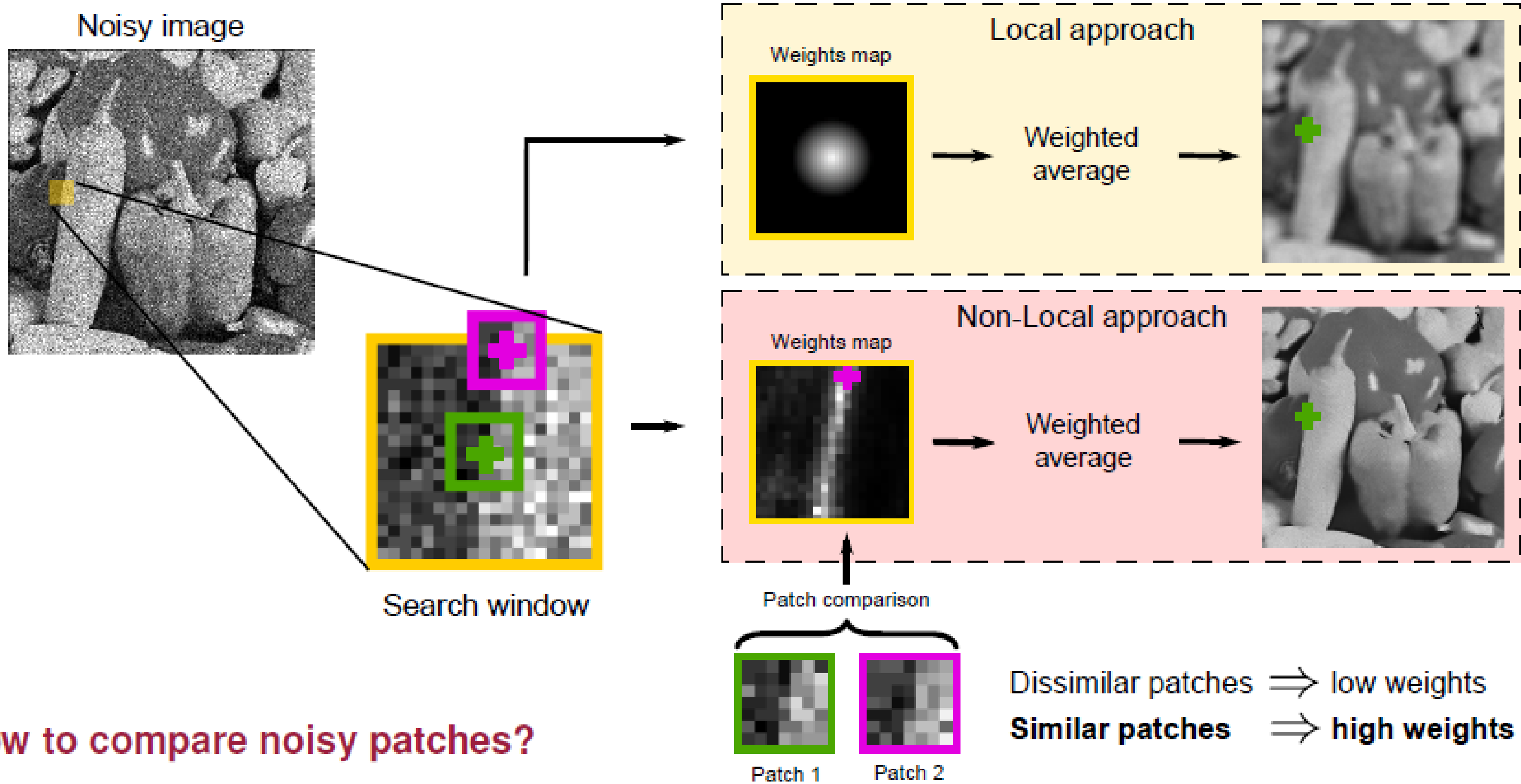
Local / non-local



Non-locality and patches



Non-locality and patches



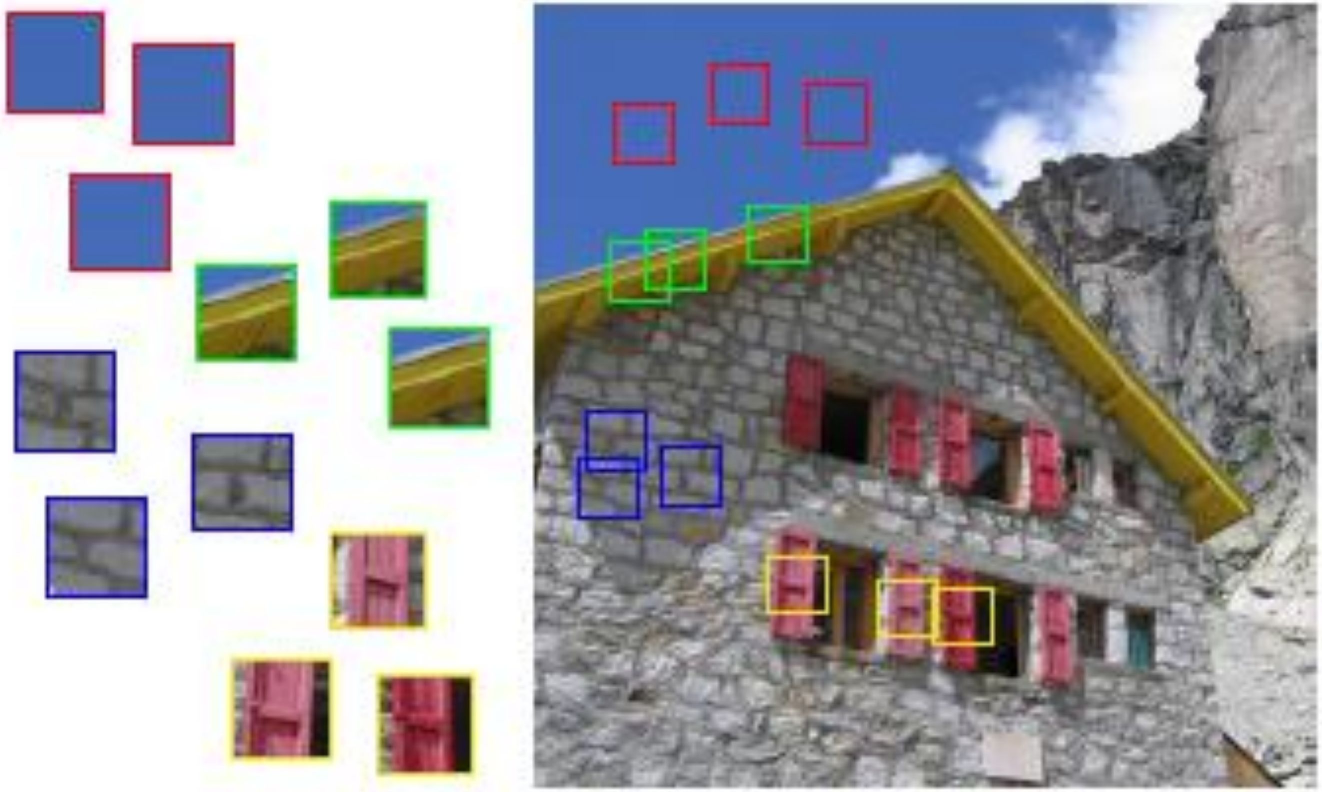
How to compare noisy patches?

Non-local means

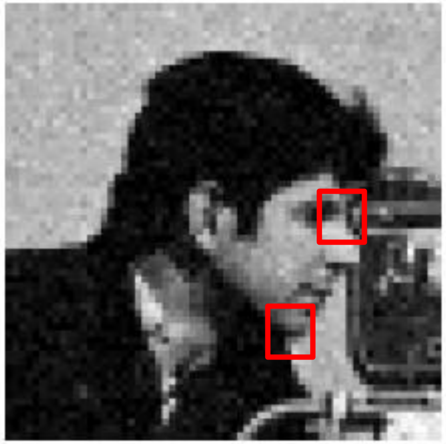
Selection based filtering – H1 redundancy

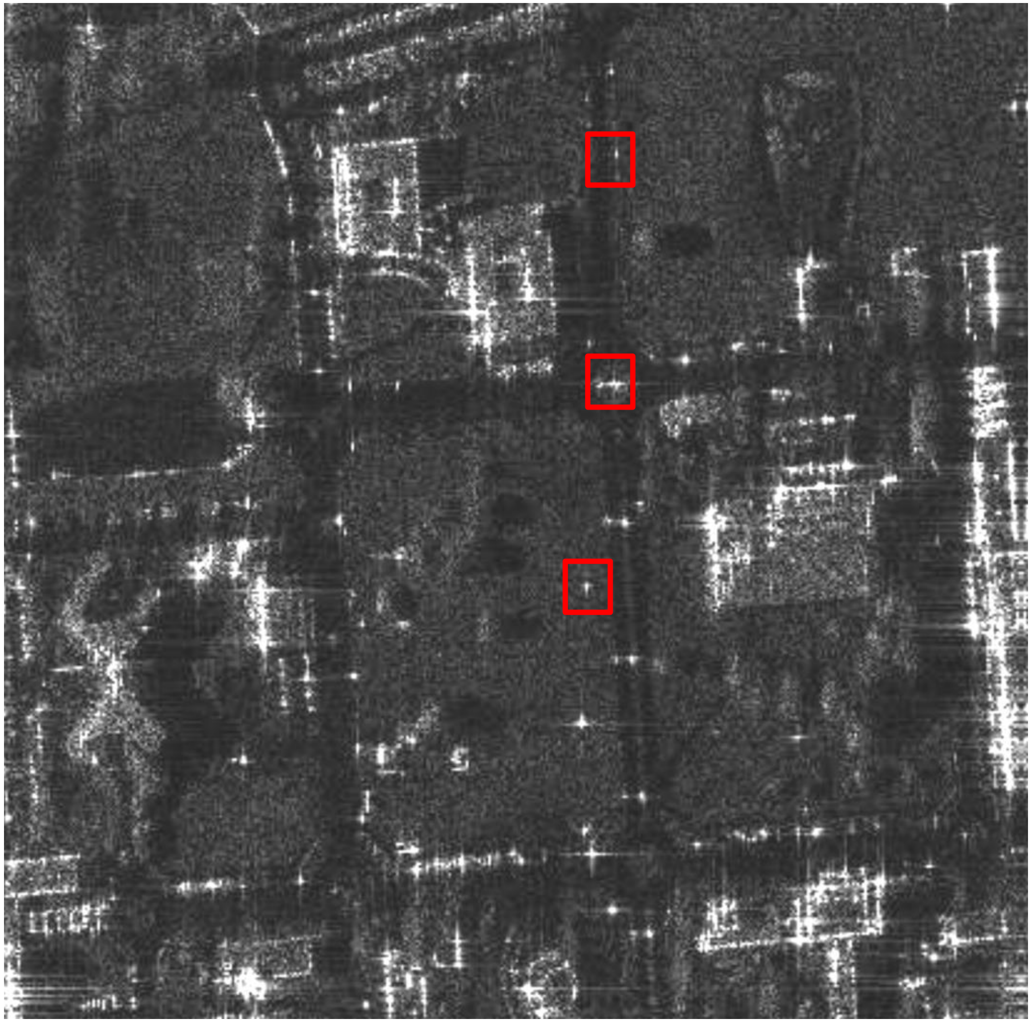
Non-local approaches - patches

H1 : Hypothesis of redundancy of patches in images



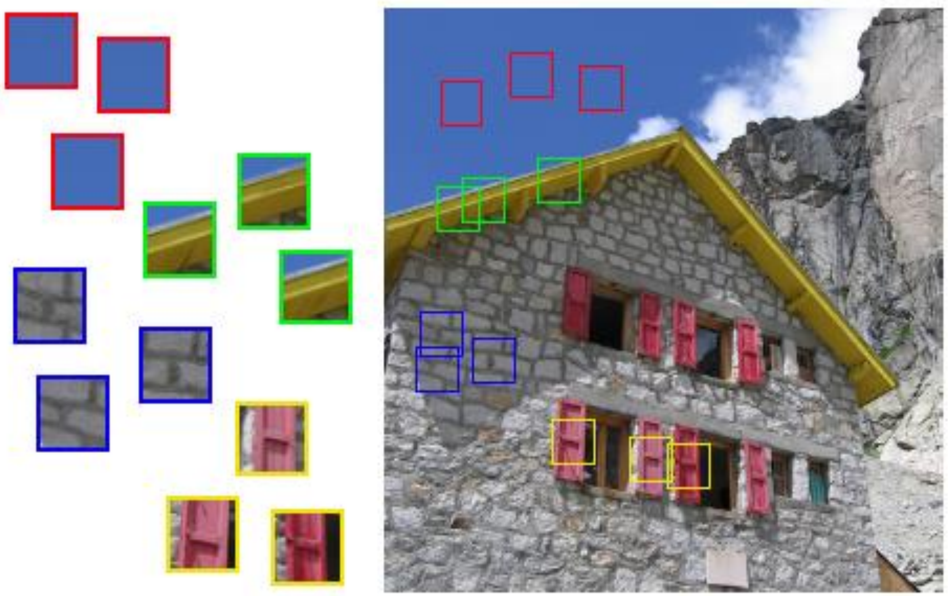
Redundancy of patches …

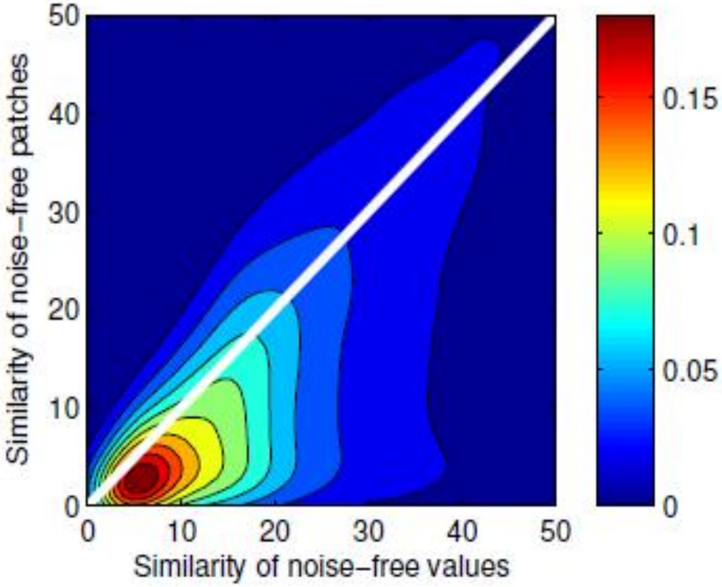




Non-local approaches

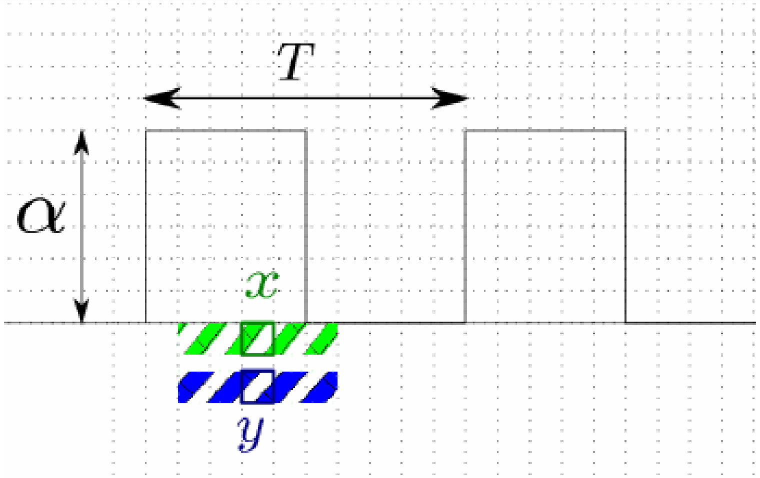
H2 : similarity between patches ➡ similarity of central pixels

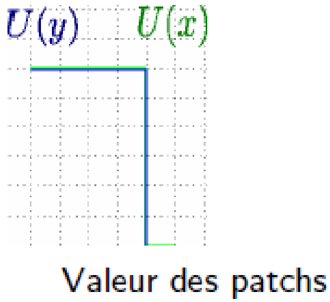




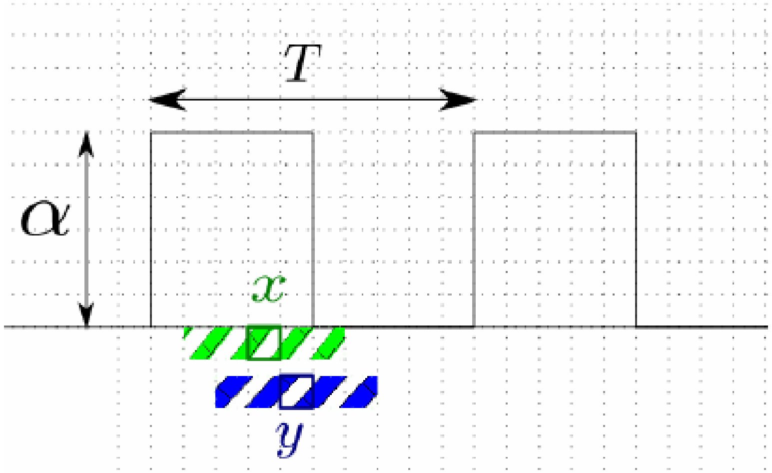
- **Introduction**
  - Denoising and models
- **Non-local / patch based approaches**
  - Principle
  - Toy examples
  - Limits and solutions
- **Advanced methods**
  - Noise adaptation
  - Iterative approaches
  - Automatic setting of parameters
  - Shape of patches

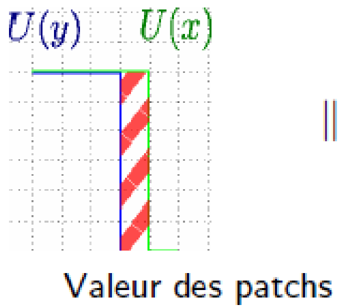
Toy examples – periodic texture





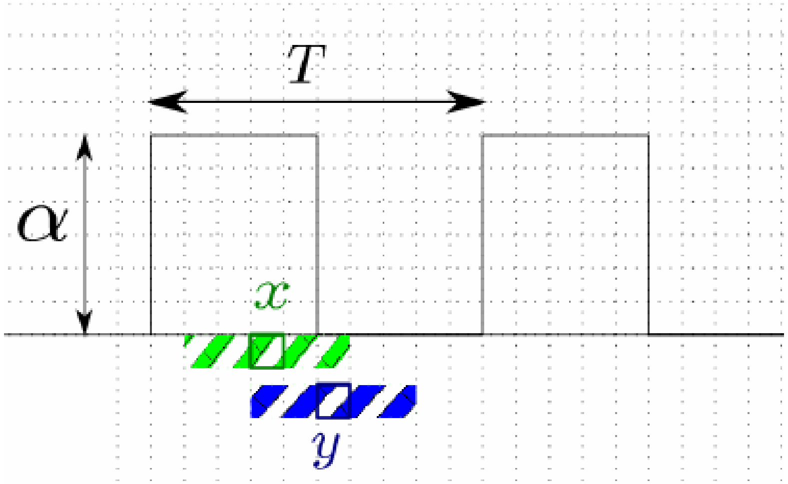
$$\|U(x) - U(x)\|^2 = 0$$

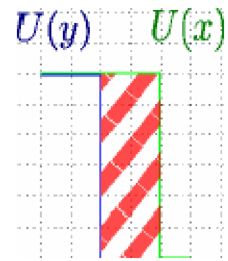




$$\|U(x) - U(x+1)\|^2 = \frac{\alpha^2}{s}$$

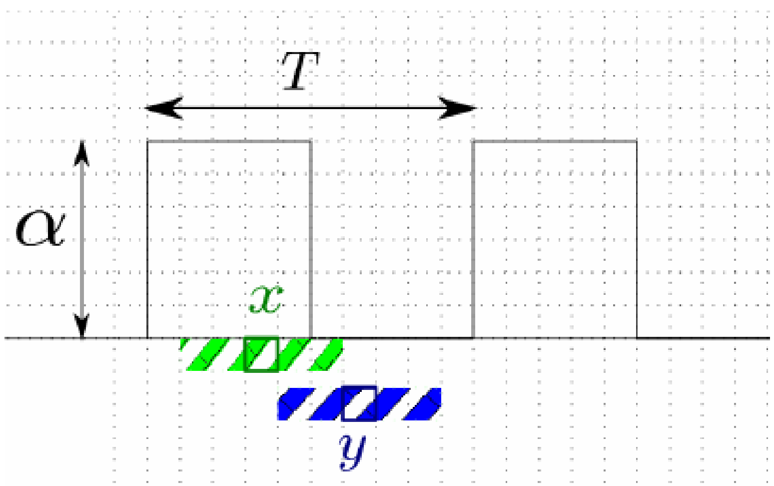
Toy examples – periodic texture

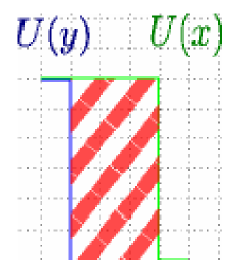




Valeur des patchs

$$\|U(x) - U(x+2)\|^2 = \frac{2\alpha^2}{s}$$

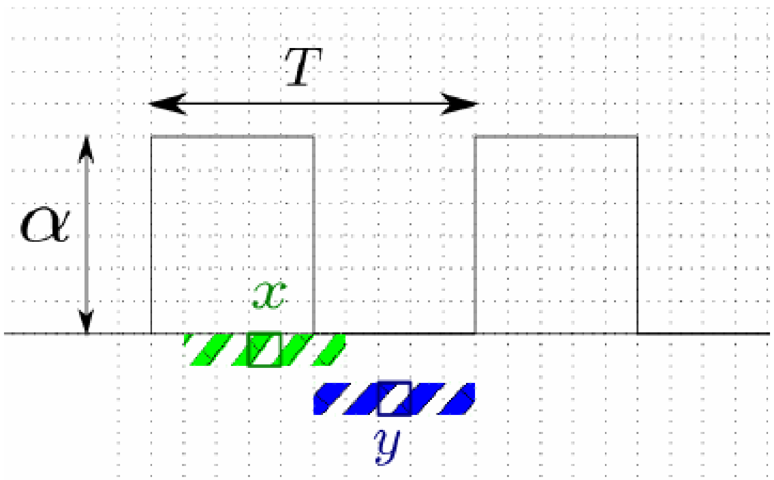


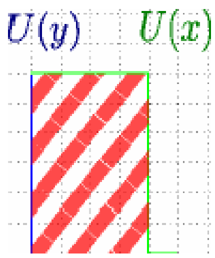


Valeur des patchs

$$\|U(x) - U(x+3)\|^2 = \frac{3\alpha^2}{s}$$

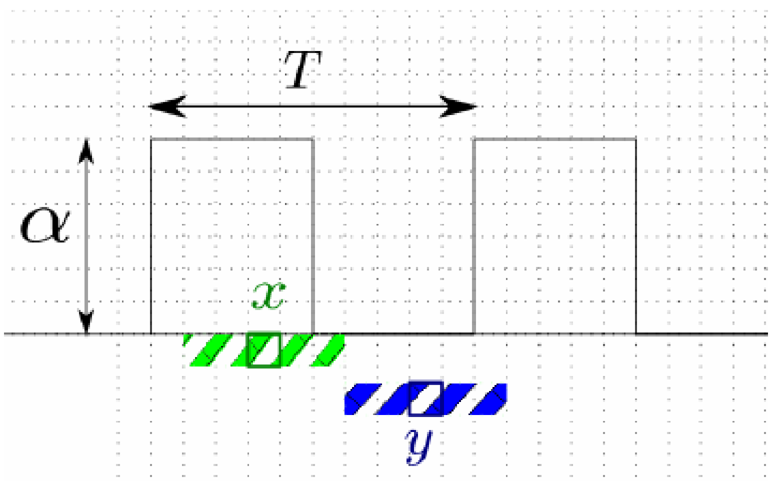
Toy examples – periodic texture

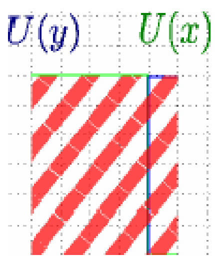




Valeur des patchs

$$\|U(x) - U(x+4)\|^2 = \frac{4\alpha^2}{s}$$

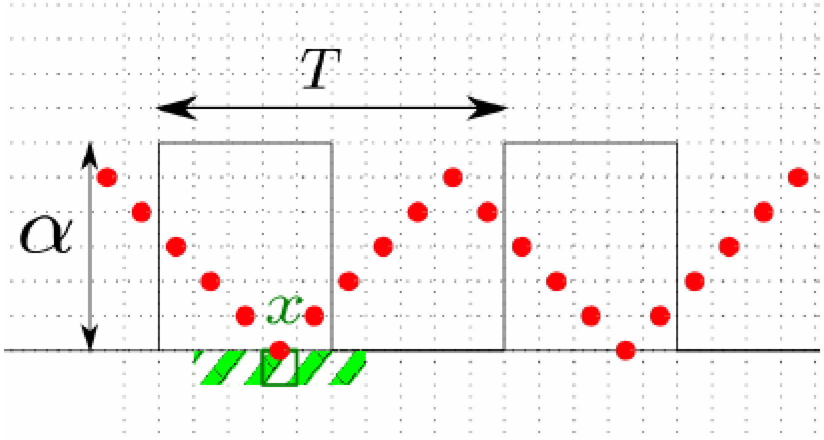




Valeur des patchs

$$\|U(x) - U(x+5)\|^2 = \frac{5\alpha^2}{s}$$

Toy examples – periodic texture

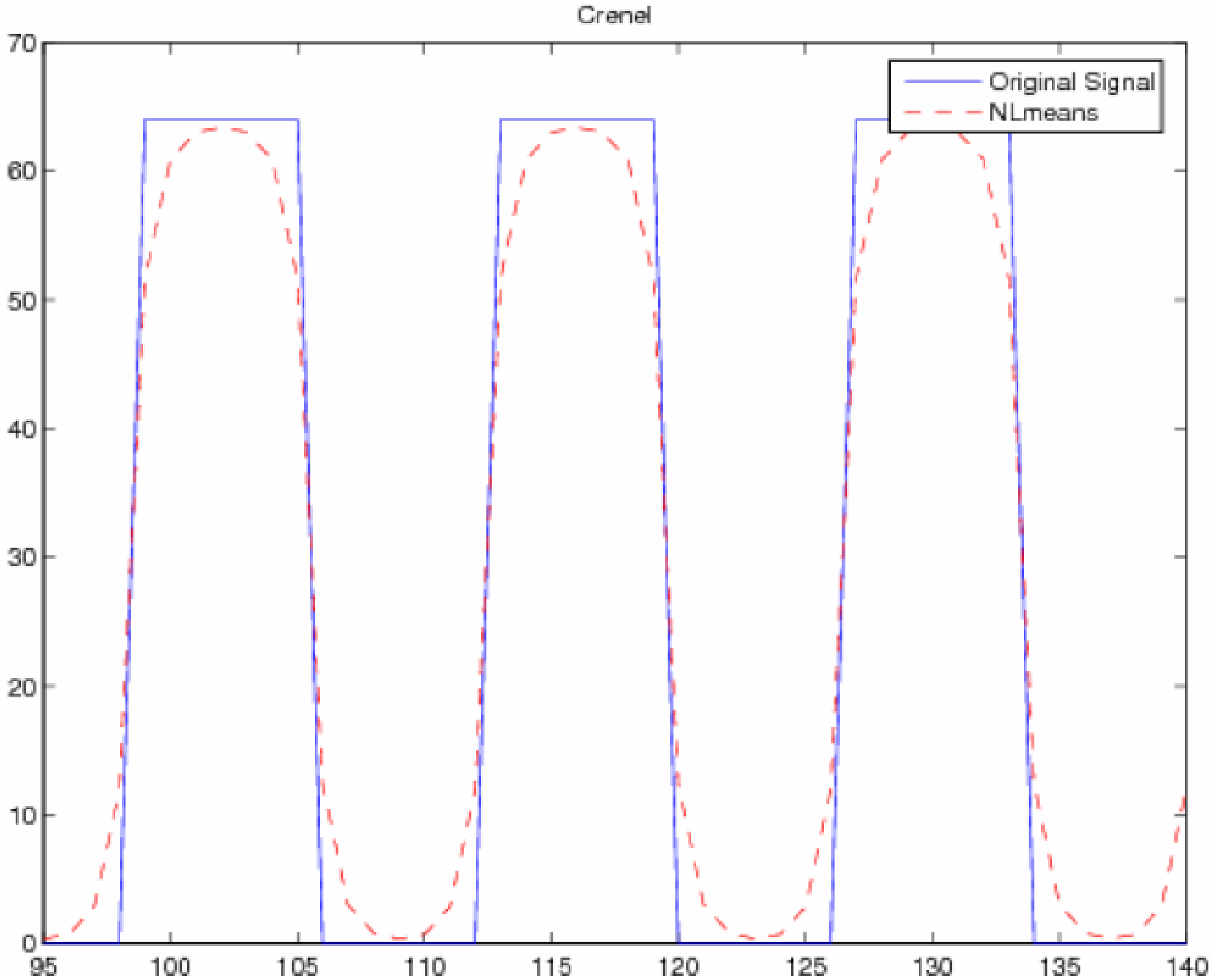


$$\|U(x) - U(x+j)\|^2 = \frac{|j|\alpha^2}{s}$$

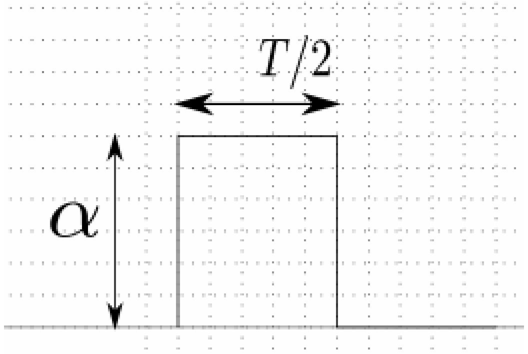
$$NLu(x) = \frac{\alpha}{(1-e^{-r\frac{T}{2}})(1+e^{-r})}\left(1 - e^{-r} - 2e^{-\frac{1}{2}(\frac{T}{2}+1)r}\cosh rx\right)$$

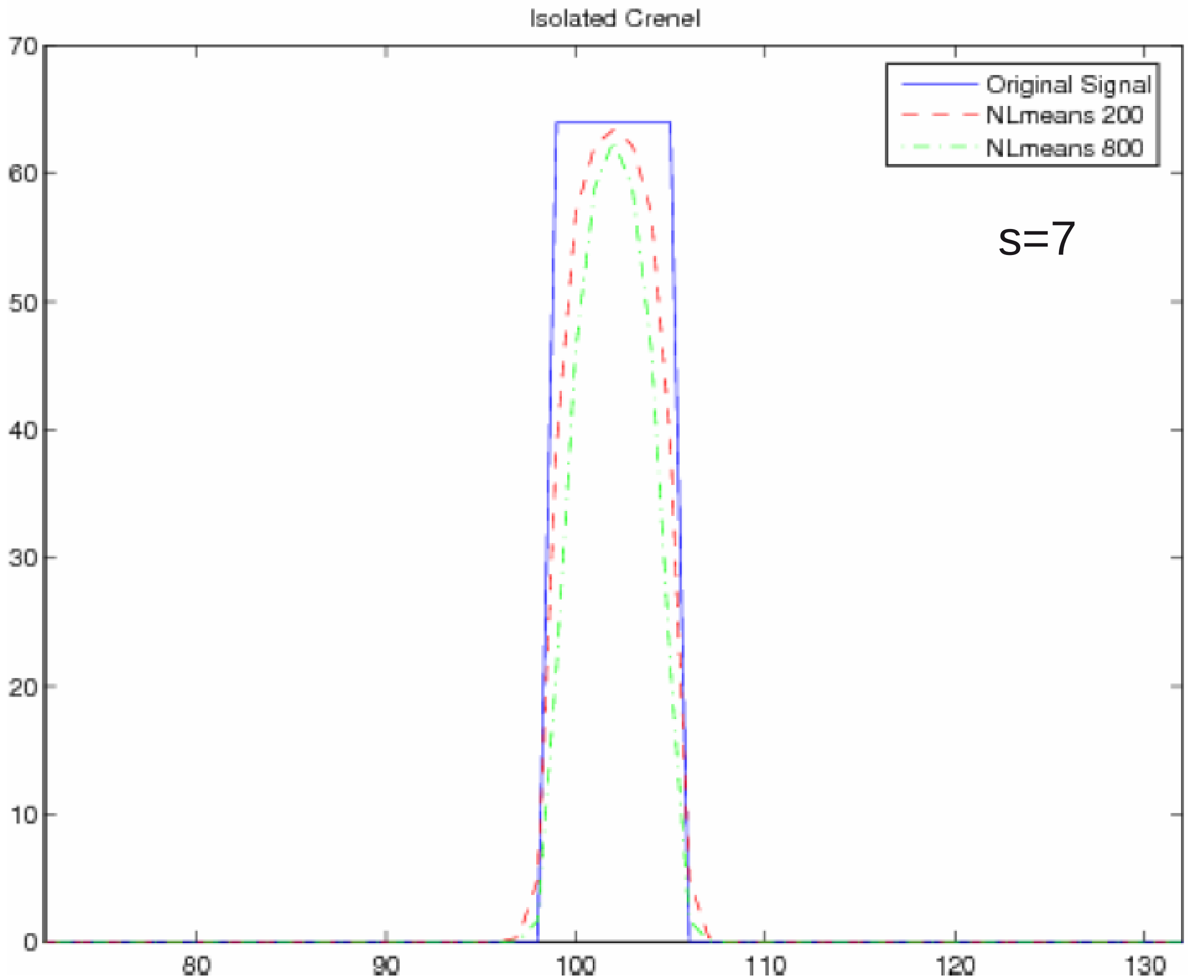
Toy examples – periodic texture

$$NLu(x) = \frac{\alpha}{(1-e^{-r\frac{T}{2}})(1+e^{-r})}\left(1 - e^{-r} - 2e^{-\frac{1}{2}(\frac{T}{2}+1)r}\cosh rx\right)$$

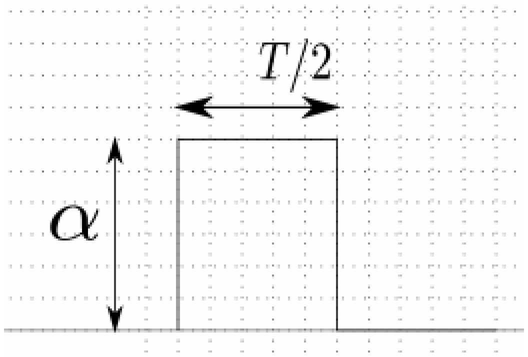


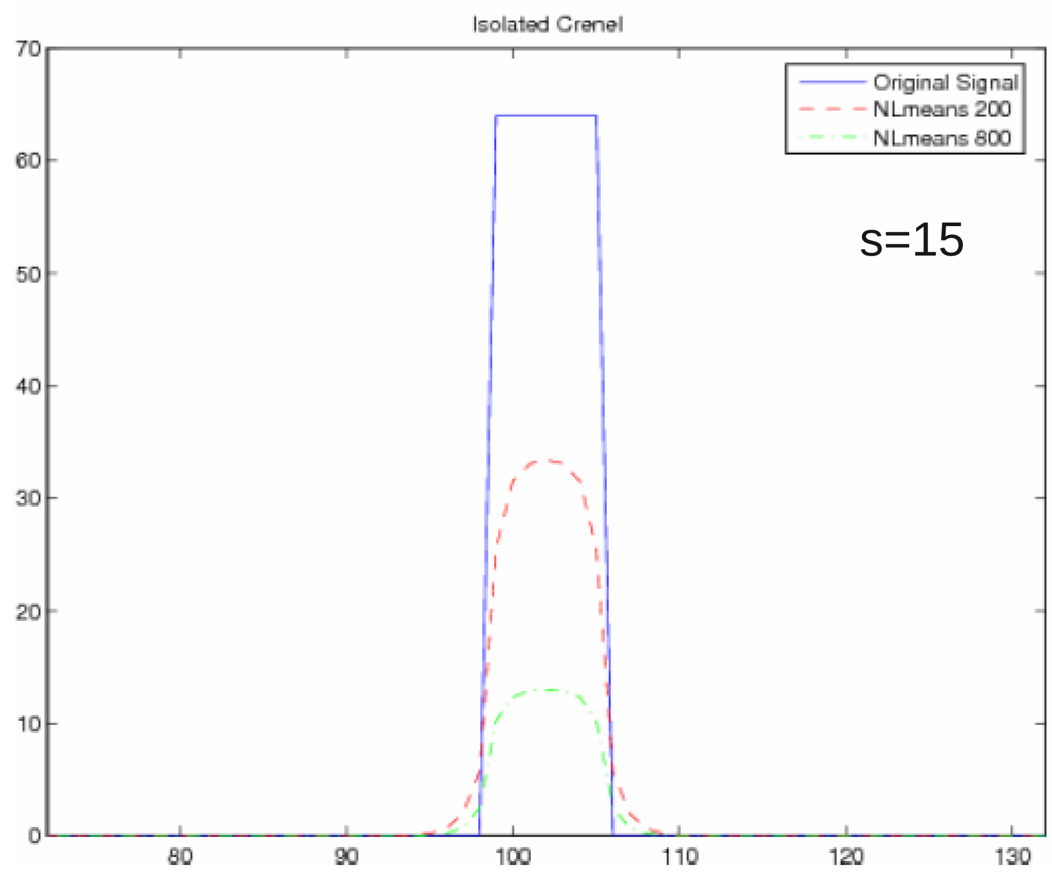
Isolated crenel





Isolated crenel

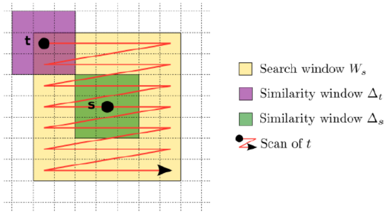




- **Introduction**
  - Denoising and models
- **Non-local / patch based approaches**
  - Principle
  - Toy examples
  - Limits and solutions
- **Advanced methods**
  - Iterative approaches
  - Automatic setting of parameters

Limits and solutions

- **Limits:**
  - Loss of weakly contrasted structures
  - « rare patch effect »: noise halo
- **Influence of NL-means parameters:**
  - Search window W
  - Patch size s
  - Kernel function (h parameter)
- **Solution:**
  - Local adaptation of h



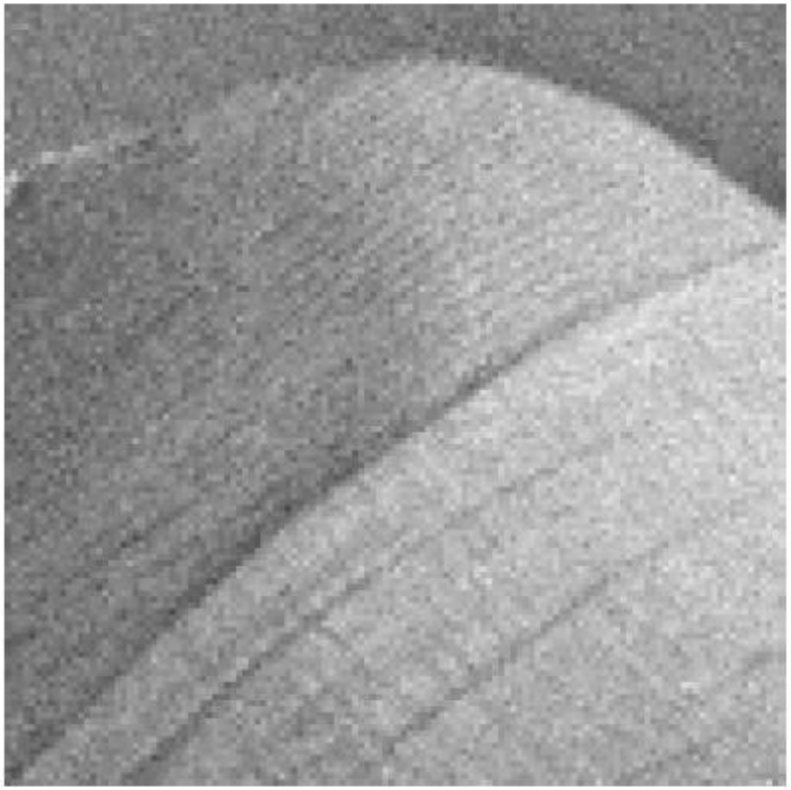
Bias / variance trade-off

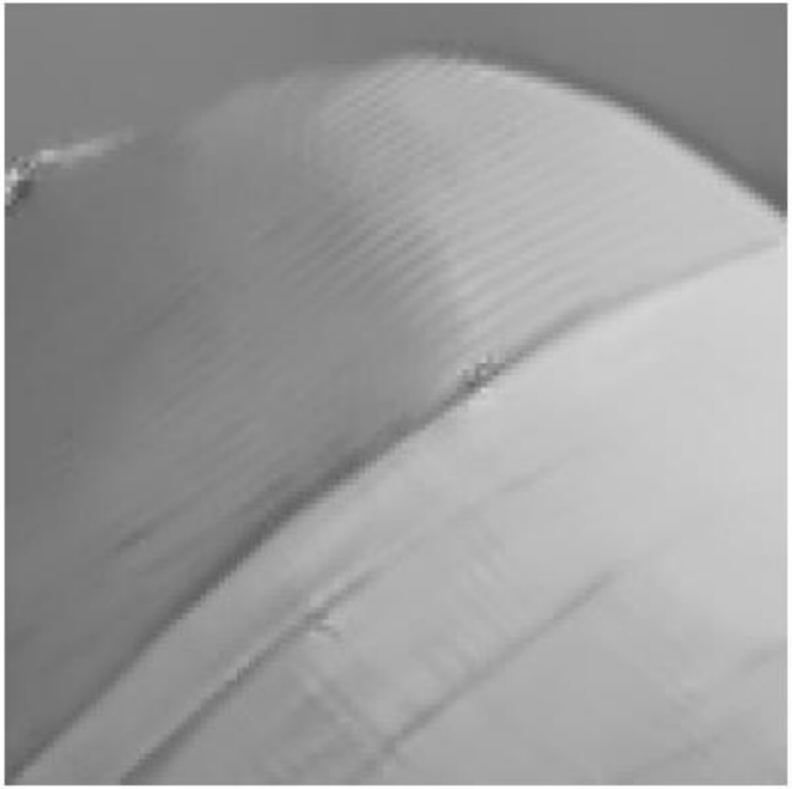
Influence of W: loss of details

W=11x11

W=61x61

Influence of W: loss of details



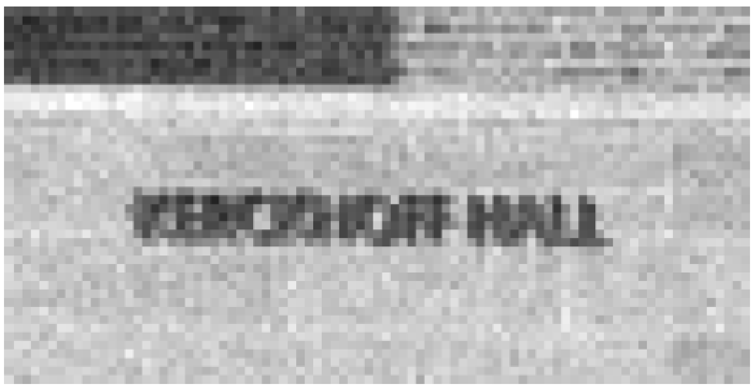


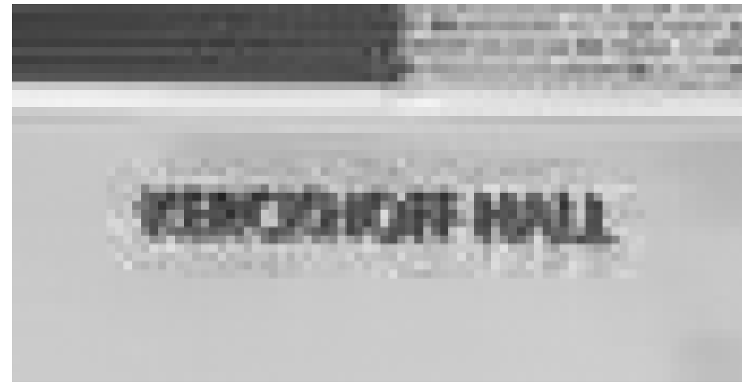
Influence of patch size: « rare patch effect »

Influence of patch size: « rare patch effect »

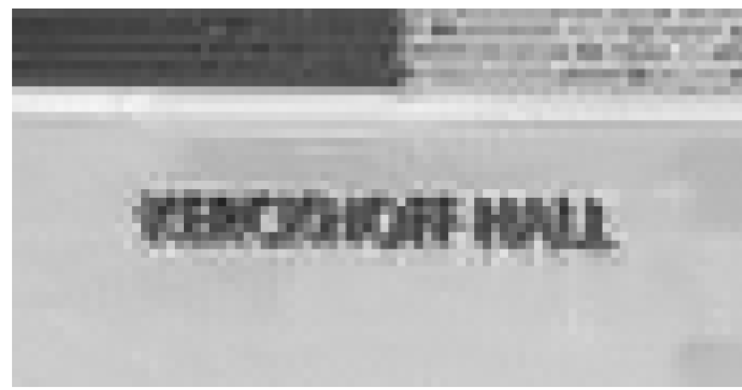
Patch 9×9

Influence of patch size





Patch 9×9



Patch 5×5

Influence of patch size


Patch 3×3

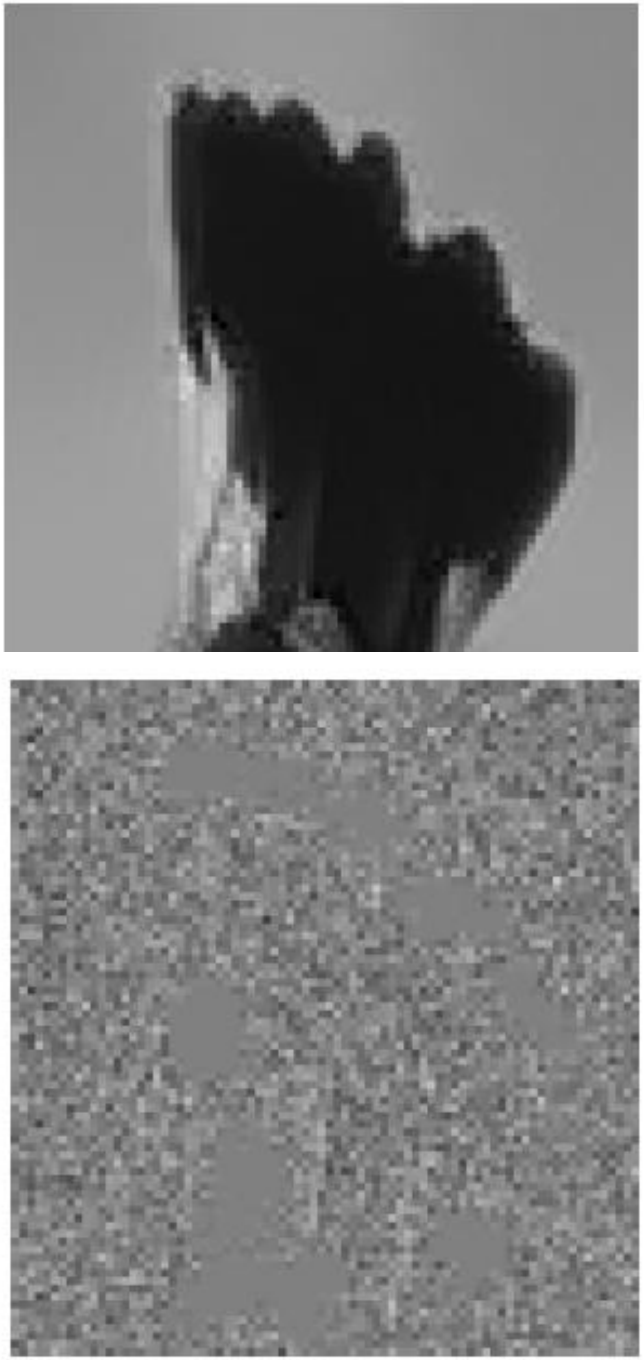

Patch 5×5

Results

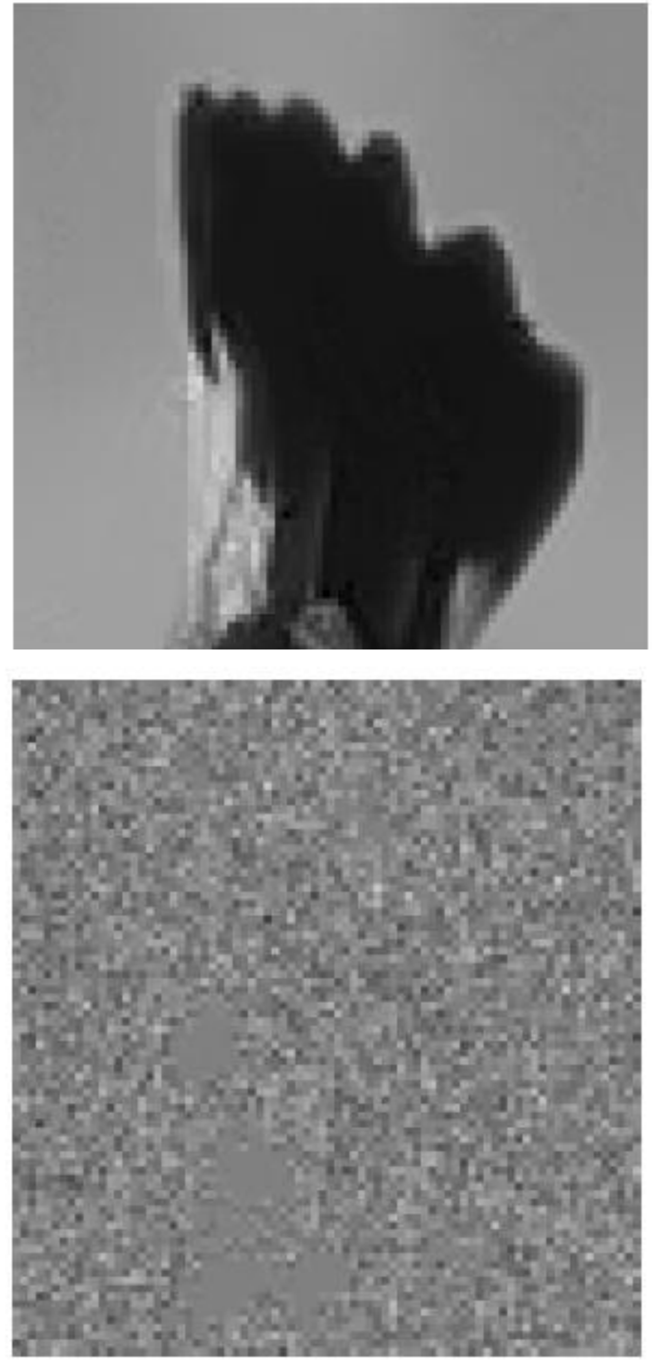
Influence of h

NLmeans, *h* global

NLmeans, *h* local

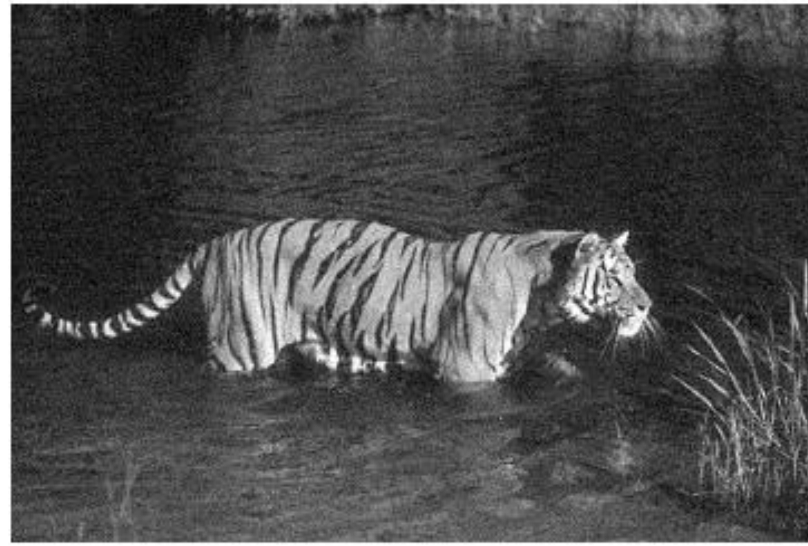


NLmeans, h global



NLmeans, h local

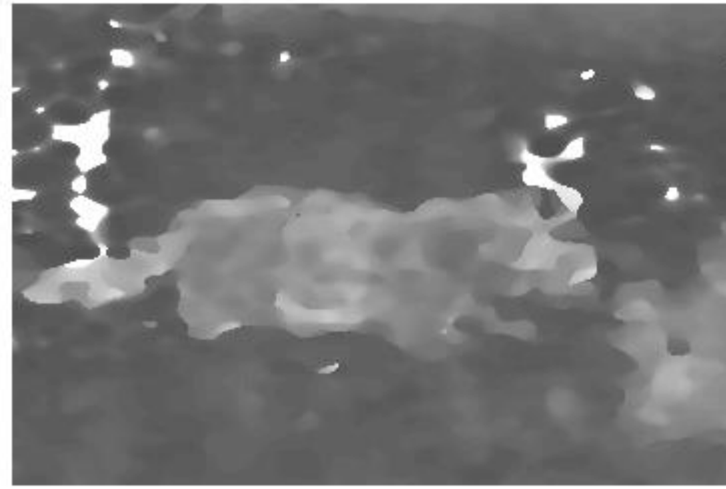
$$PSNR(\hat{u}, u) = 10 \log_{10} \frac{255^2}{\frac{1}{N} \|\hat{u} - u\|_2^2}$$



NLmeans, *h* global (PSNR 31.71 dB)

NLmeans, *h* local (PSNR 32.33 dB)

h adaptation



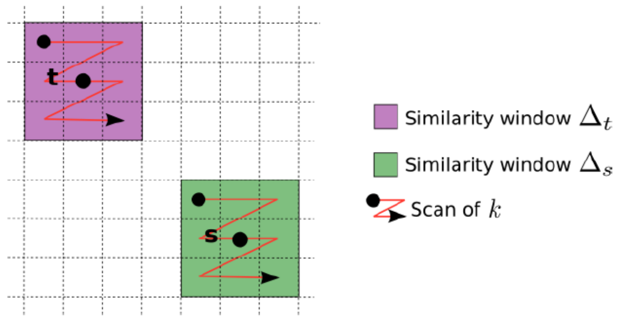
NLmeans, h global

NLmeans, h local

- **Introduction**
  - Denoising and models
- **Non-local / patch based approaches**
  - Principle
  - Toy examples
  - Limits and solutions
- **Advanced methods**
  - Noise adaptation
  - Iterative approaches
  - Automatic setting of parameters
  - Shape of patches

How to compare noisy patches ?

- **Buades et al. (2005)**
  - Euclidean distance between patches
  - Implicit assumption of AWGN



when $u_1 = u_2$:

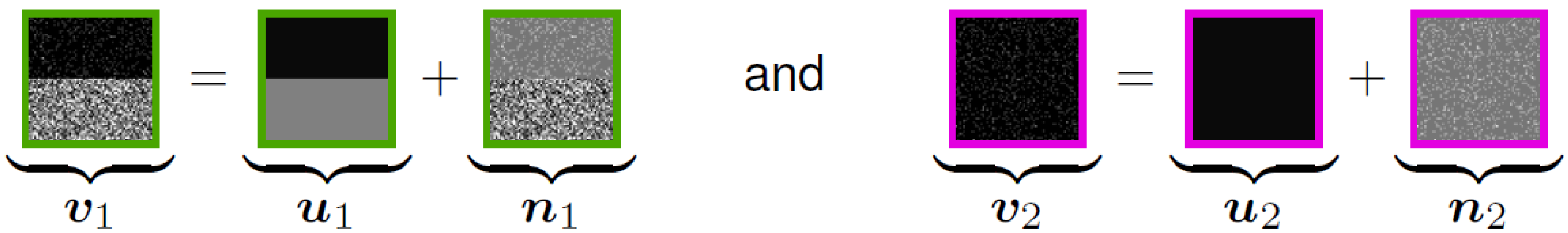
is low $\Rightarrow$ decide “similar”

when $u_1 \neq u_2$:

is high $\Rightarrow$ decide “dissimilar”

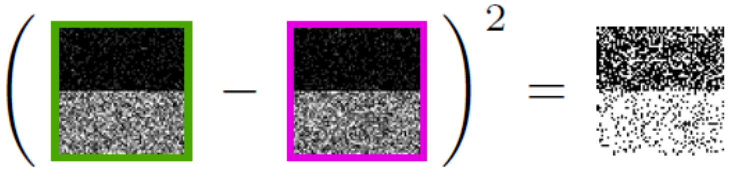
How to compare noisy patches ?

- **Example of signal dependant-noise:**

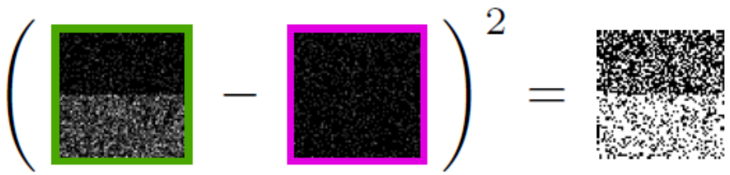


- Limits of the euclidean distance:

when $u_1 = u_2$:



when $u_1 \neq u_2$:



How to compare noisy patches ?

Noisy image
(gaussian noise)

Denoised (« oracle »
Driven by noise-free
Image content)

Denoised
(driven by noisy
Image content)

TELECOM ParisTech

How to compare noisy patches ?

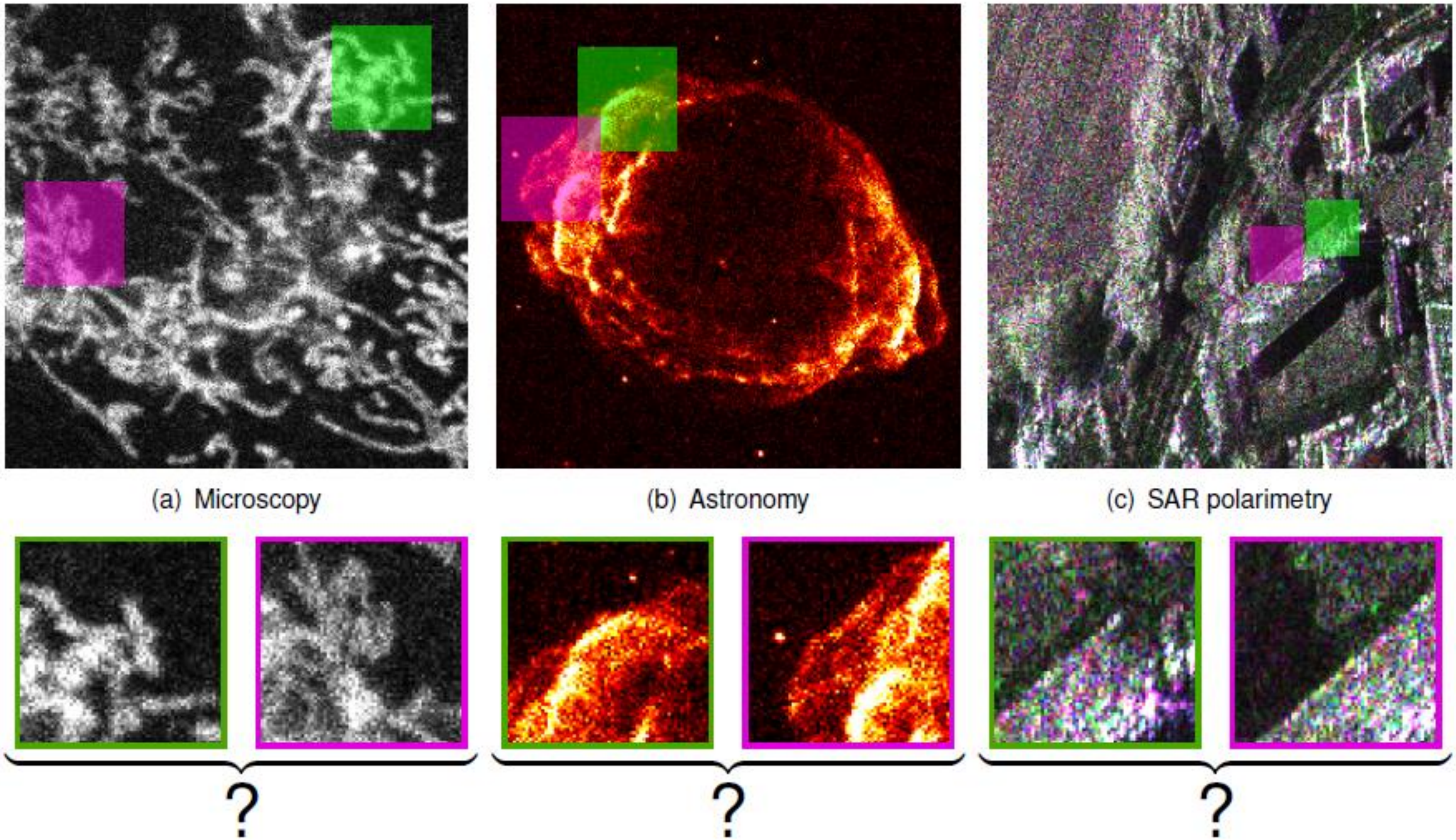
Noisy image
(Poisson noise
Signal dependent noise)

Denoised (« oracle »
driven by noise-free
Image content)

Denoised
(driven by noisy
Image content)

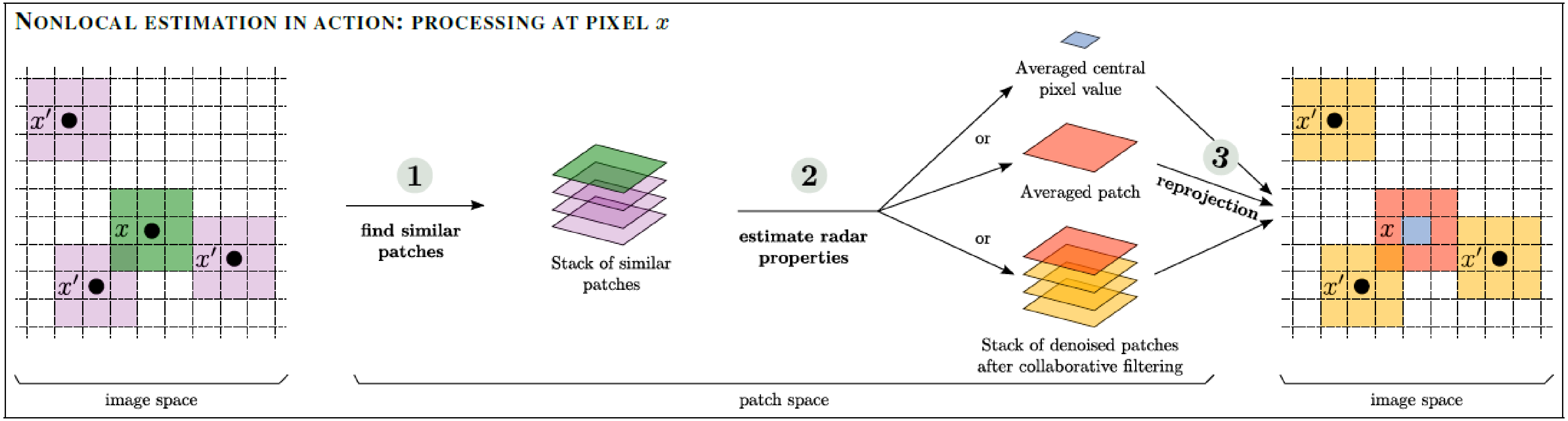
Noise distribution has to be taken into account

How to compare noisy patches ?



How to take into account the noise model?

Steps of non-local denoising



A probabilistic framework

- **Principle: adaptation of the NL-means to any kind of (known) noise distribution**
  - Estimation step:

  Weighted average is replaced by weighted maximum likelihood estimation

$$\boxed{\hat{u}(x) = \arg\max_{t} \sum_{x'} w(x,x') \log p(v(x')|t)}$$

  - Detection of similar patches:

  Weight definition is defined in a detection framework by *hypothesis testing*

Similarity definition

- **Similarity is defined by an hypothesis test:**

$$\mathcal{H}_0 : u_1 = u_2 \equiv u_{12} \qquad \text{(null hypothesis)}$$

$$\mathcal{H}_1 : u_1 \neq u_2 \qquad \text{(alternative hypothesis)}$$

- **Performance measured by:**

$$P_{FA} = \mathbb{P}(\text{decide “dissimilar”} \mid u_{12}, \mathcal{H}_0) \qquad \text{(false-alarm rate)}$$

$$P_D = \mathbb{P}(\text{decide “dissimilar”} \mid u_1, u_2, \mathcal{H}_1) \qquad \text{(detection rate)}$$

- **The likelihood ratio test maximizes PD**

$$L(v_1, v_2) = \frac{p(v_1, v_2 \mid u_{12}, \mathcal{H}_0)}{p(v_1, v_2 \mid u_1, u_2, \mathcal{H}_1)}$$

Similarity definition

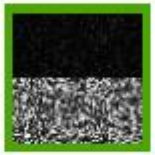
- **Unknown values are replaced by ML estimates (GLR):**

$$\frac{\sup_t p(v_1, v_2 \mid u_{12} = t, \mathcal{H}_0)}{\sup_{t_1, t_2} p(v_1, v_2 \mid u_1 = t_1, u_2 = t_2, \mathcal{H}_1)}$$

$$\frac{p(v_1 \mid u_1 = \hat{t}_{12})\; p(v_2 \mid u_2 = \hat{t}_{12})}{p(v_1 \mid u_1 = \hat{t}_1)\; p(v_2 \mid u_2 = \hat{t}_2)}$$

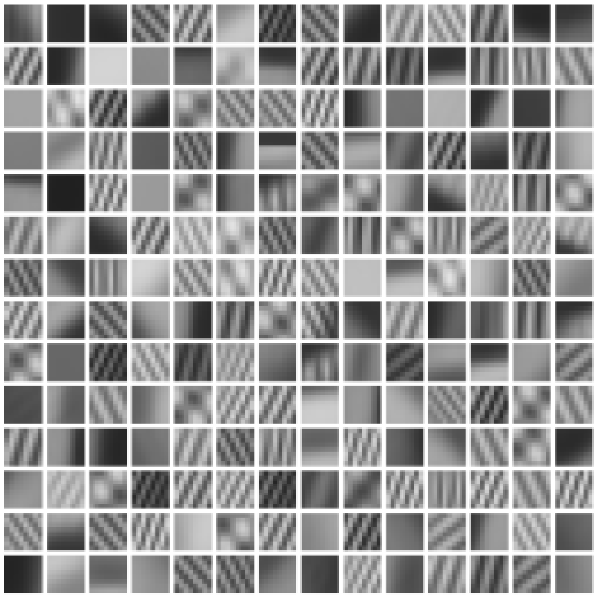
- **Study of this criterion**

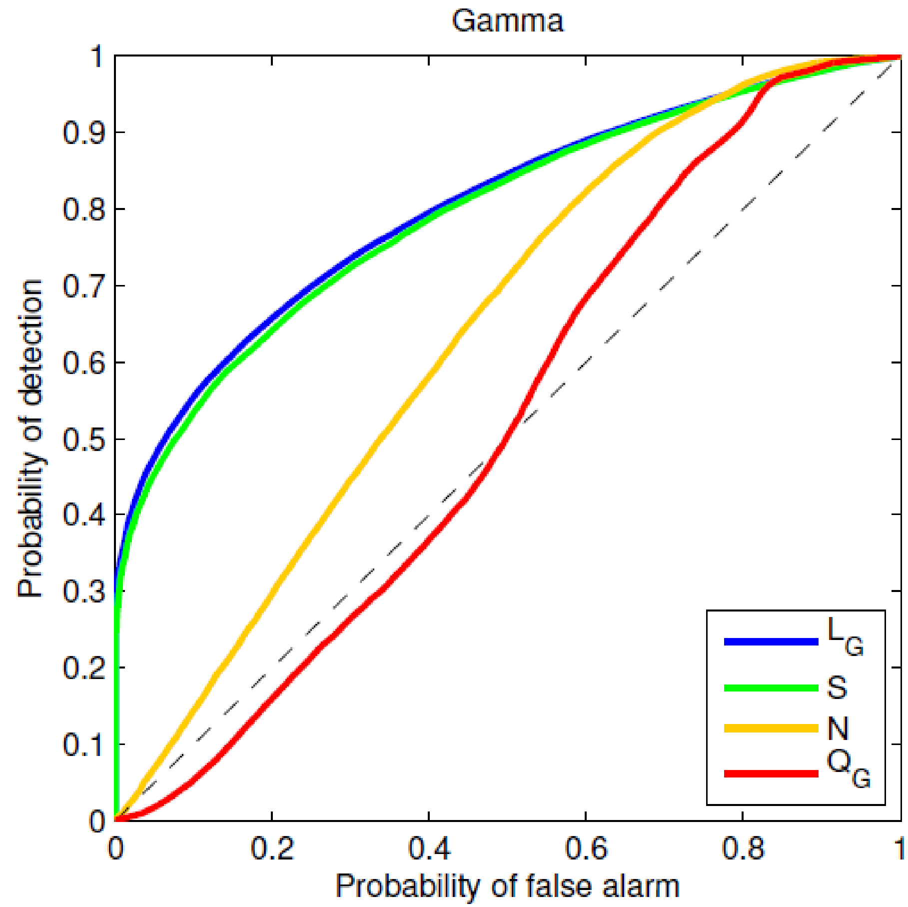
GLR

when $u_1 = u_2$: $-\log GLR$ ( , ) = 

when $u_1 \neq u_2$: $-\log GLR$ ( , ) = 

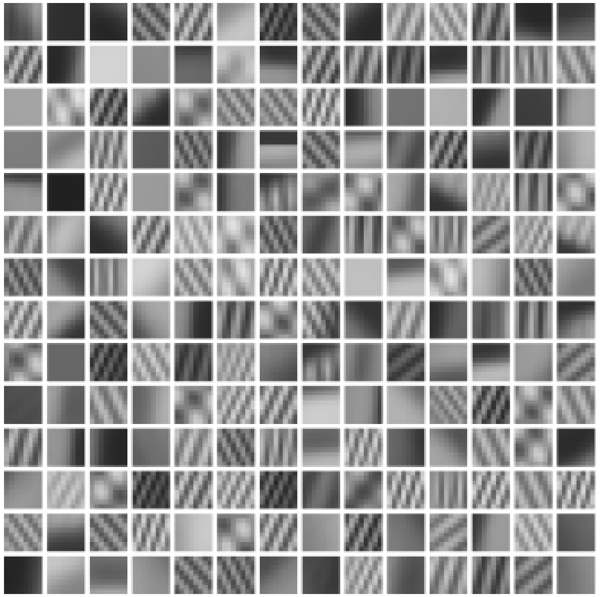
Evaluation of similarity criterion

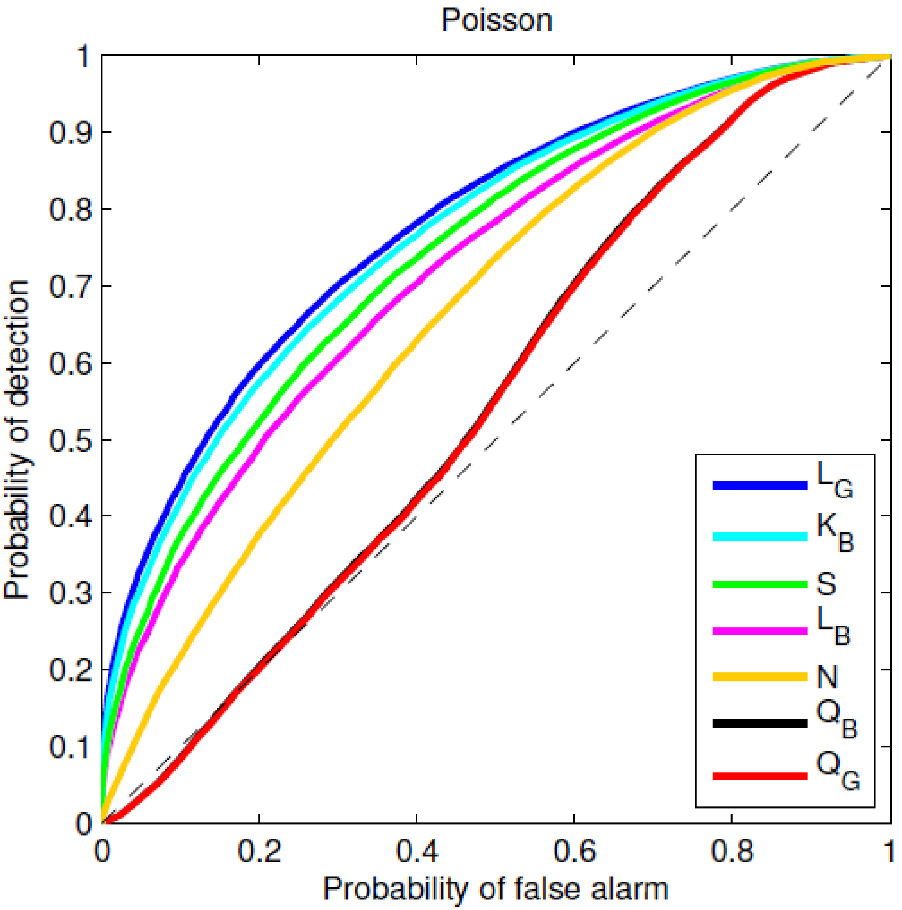




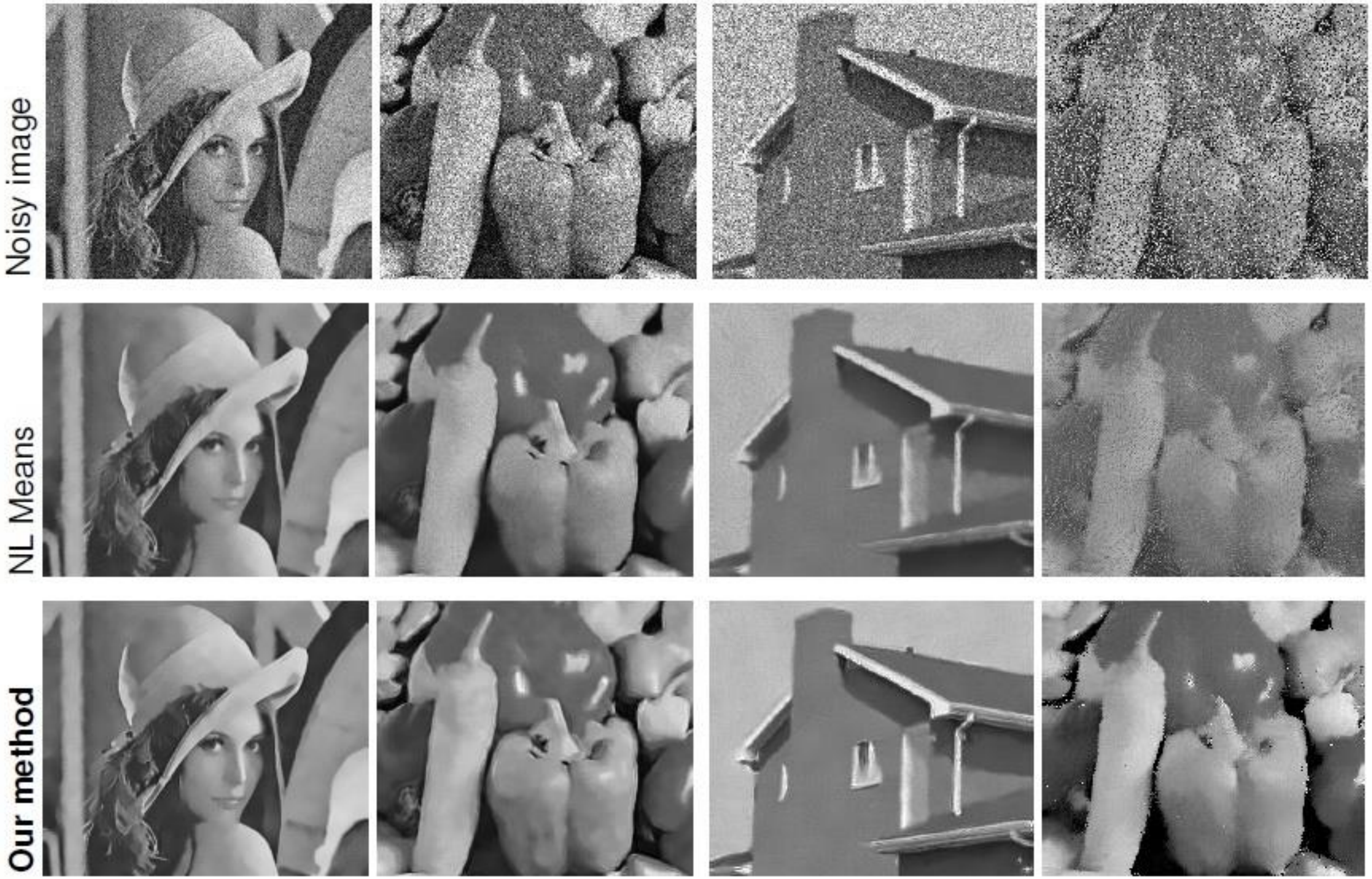
- Generalized likelihood ratio
- Variance stabilization
- Euclidean distance
- Maximum joint likelihood [Alter et al., 2006]
- Mutual information kernel [Seeger, 2002]
- Bayesian likelihood ratio [Minka, 1998, Minka, 2000]
- Bayesian joint likelihood [Yianilos, 1995, Matsushita and Lin, 2007]

Evaluation of similarity criterion





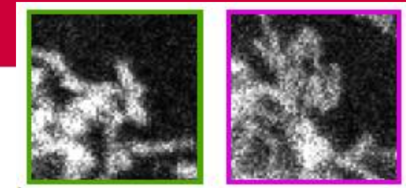
- Generalized likelihood ratio [Alter et al., 2006]
- Variance stabilization
- Euclidean distance
- Maximum joint likelihood
- Mutual information kernel [Seeger, 2002]
- Bayesian likelihood ratio [Minka, 1998, Minka, 2000]
- Bayesian joint likelihood [Yianilos, 1995, Matsushita and Lin, 2007]



(a) Gaussien +0.87 dB (b) Poisson +1.13 dB (c) *Speckle* +4.00 dB (d) Impuls. +3.82 dB

- **Introduction**
  - Denoising and models
- **Non-local / patch based approaches**
  - Principle
  - Toy examples
  - Limits and solutions
- **Advanced methods**
  - Noise adaptation
  - Iterative approaches
  - Automatic setting of parameters
  - Shape of patches

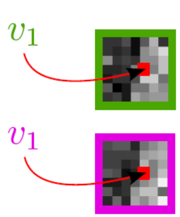
Iterative version approaches
Similarity definition - refinement

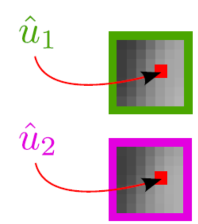


$$\frac{\mathbb{P}(\mathcal{H}_0 |\ {}_1,\ {}_2)}{\mathbb{P}(\mathcal{H}_1 |\ {}_1,\ {}_2)} = \boxed{\frac{p(\ {}_1,\ {}_2 | \mathcal{H}_0)}{p(\ {}_1,\ {}_2 | \mathcal{H}_1)}} \times \boxed{\frac{\mathbb{P}(\mathcal{H}_0)}{\mathbb{P}(\mathcal{H}_1)}}$$

Computed on noisy data
using noise distribution and GLR

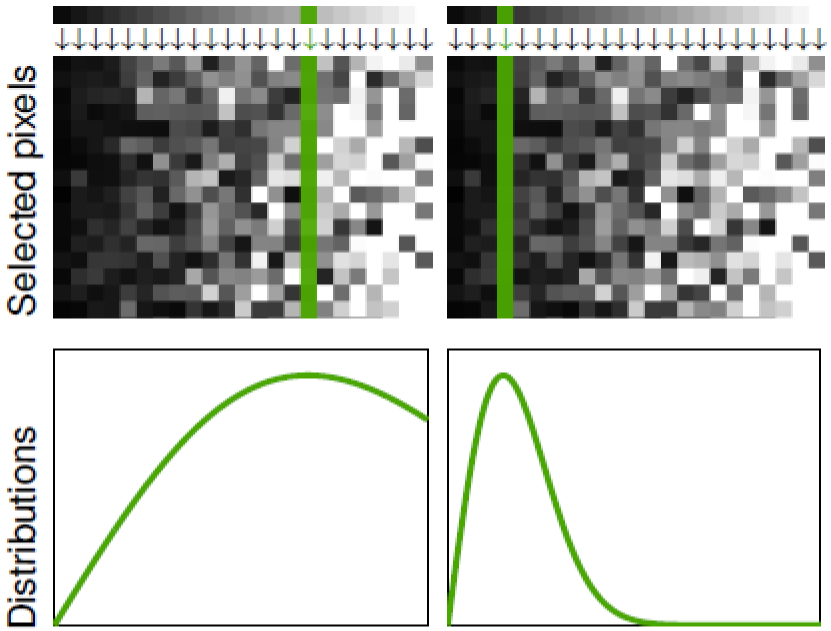
Computed on noise-free data
using an iterative scheme
and symmetrical KL divergence



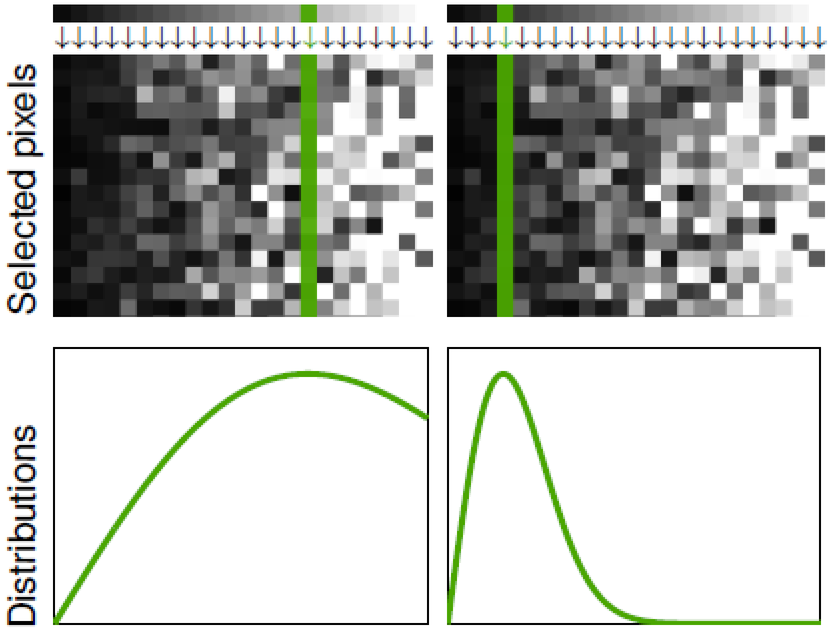


$$\mathcal{D}_{KL}(\hat{u}_1 \| \hat{u}_2)$$

Iterative version- Weight refinement

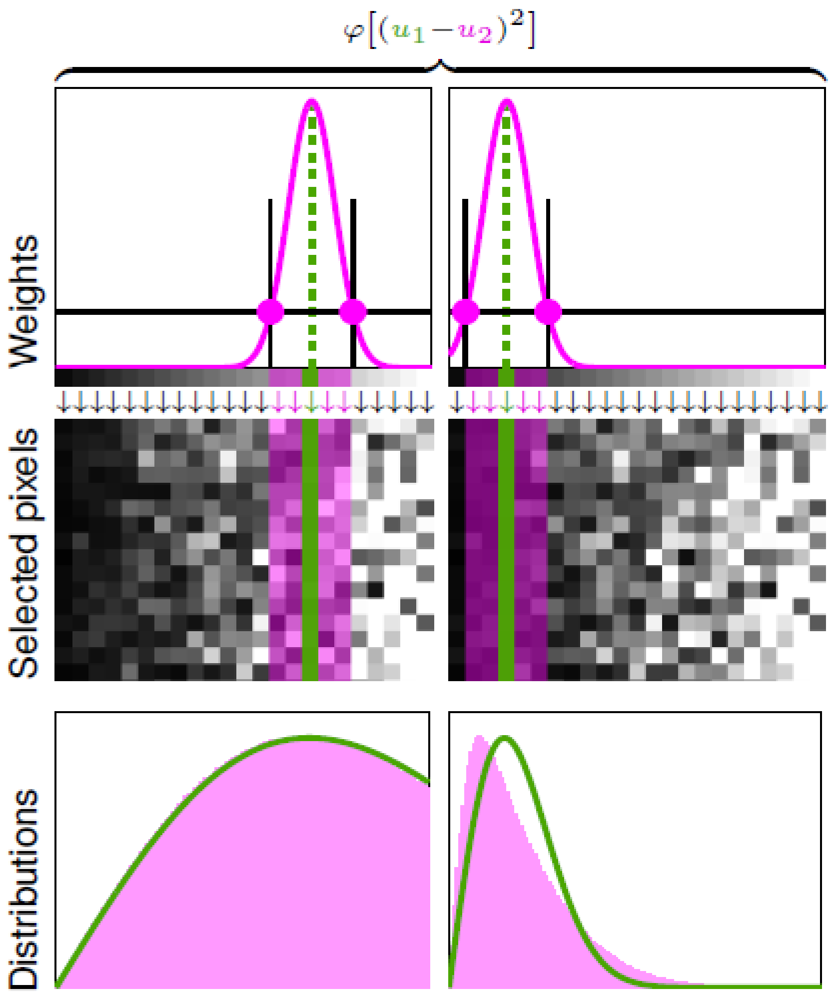


Iterative version- Weight refinement

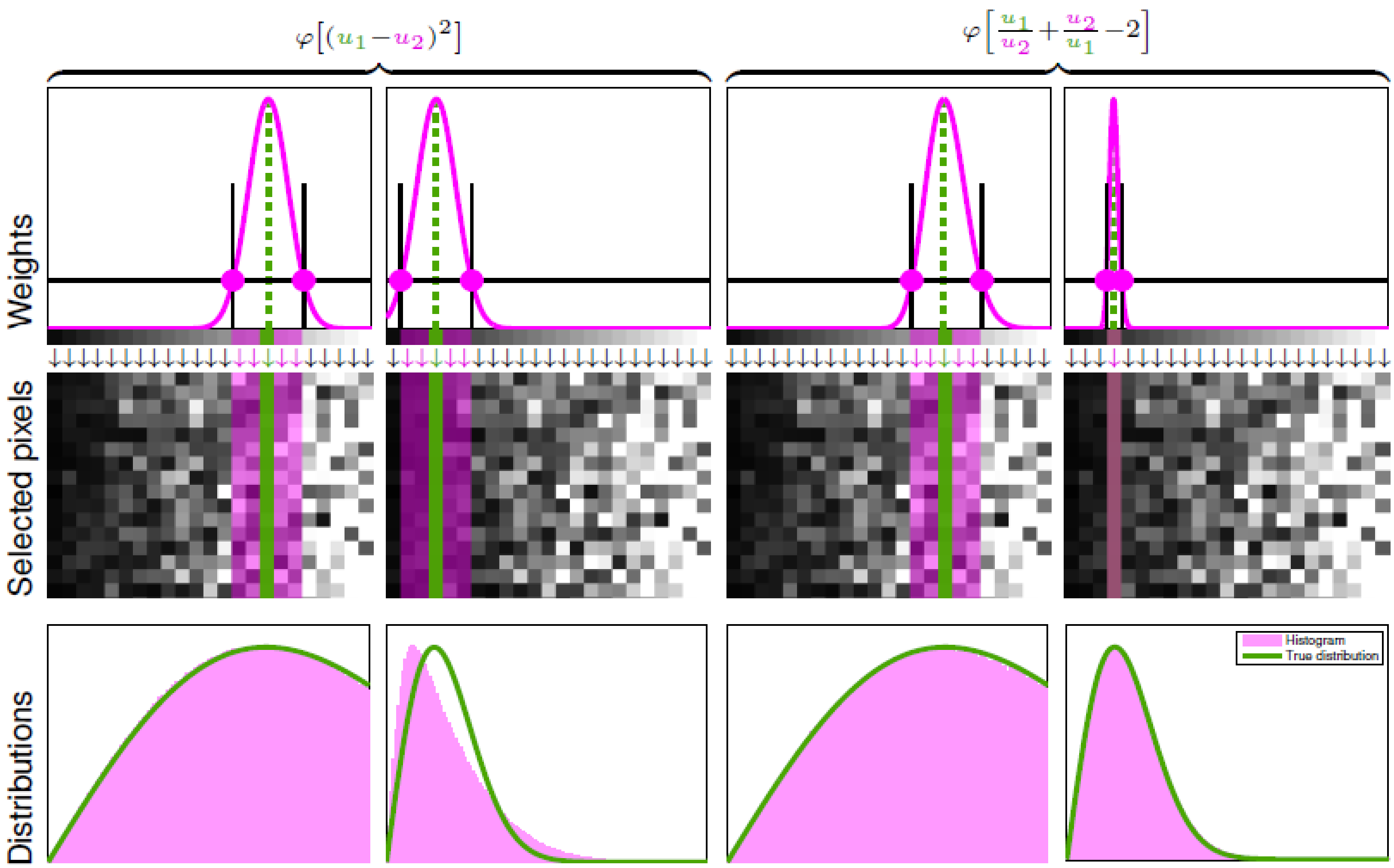


TELECOM ParisTech

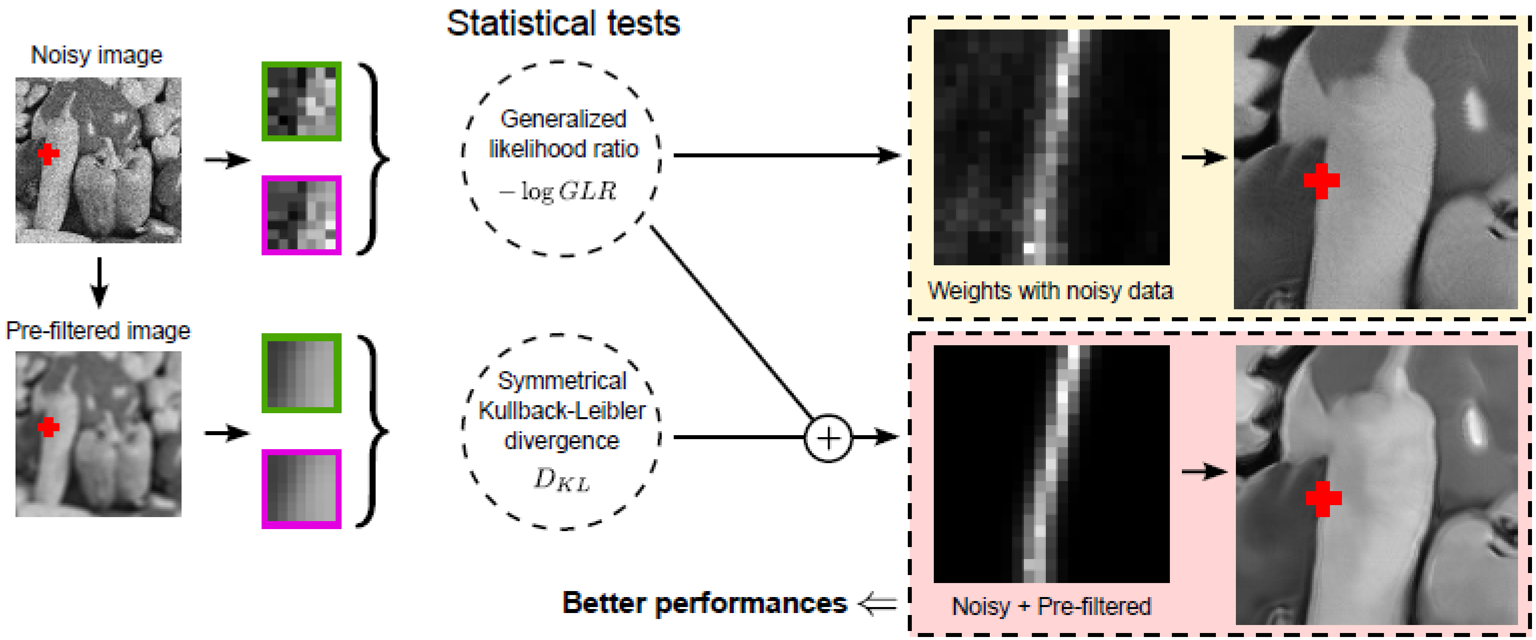
Iterative version - Weight refinement



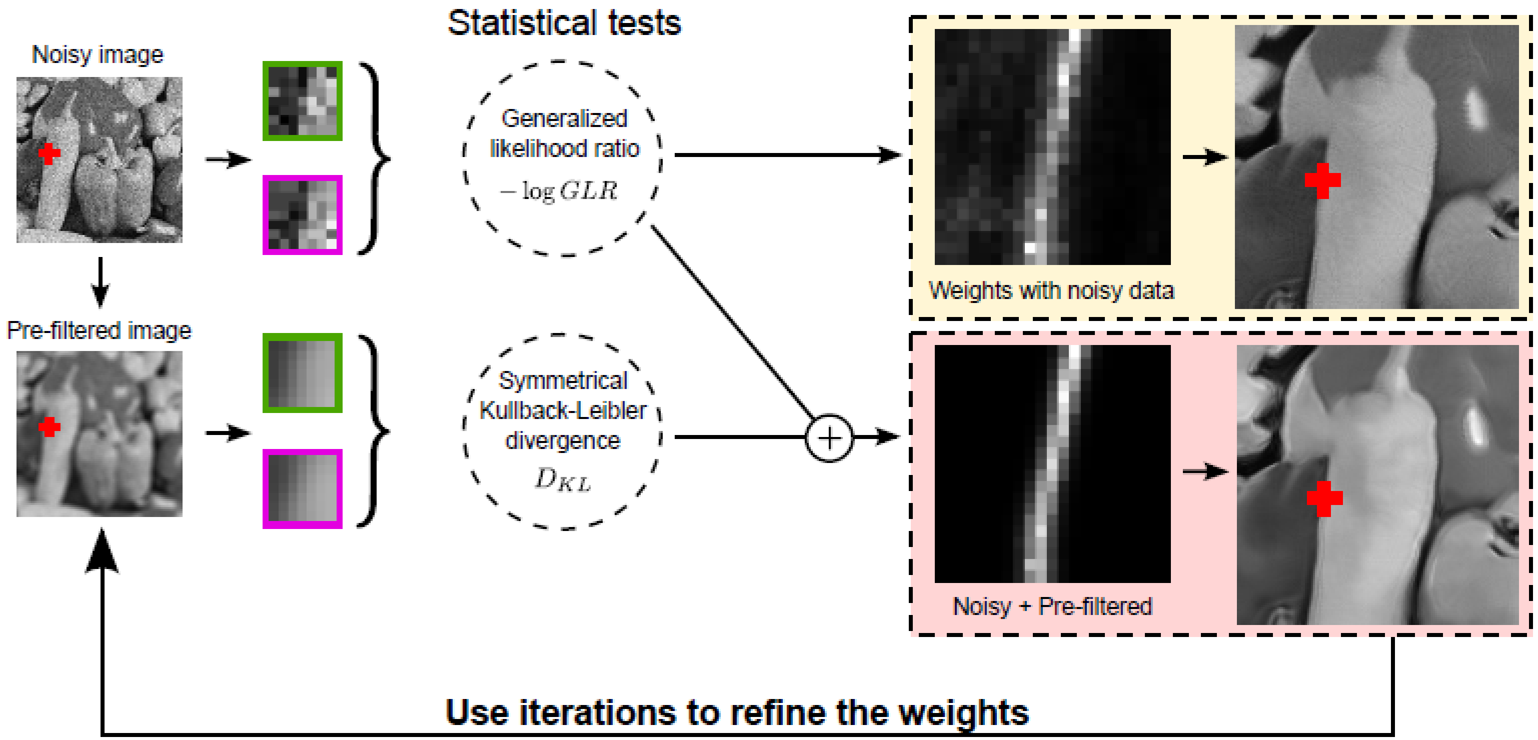
Iterative verion - Weight refinement



Iterative verion - Global scheme



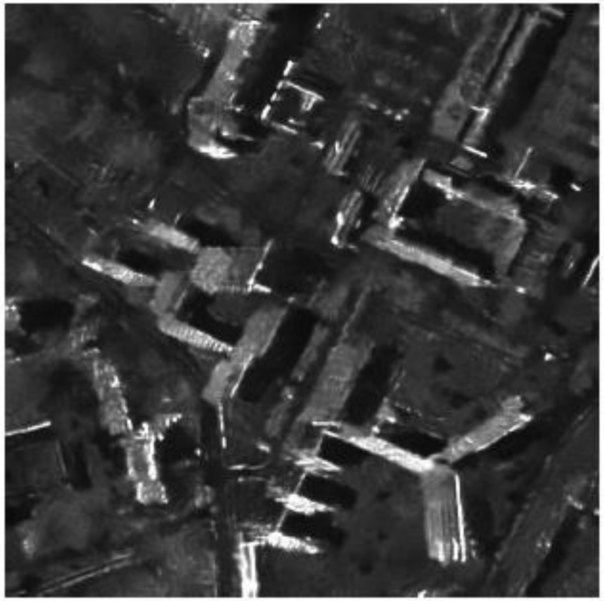
Iterative version - Global scheme



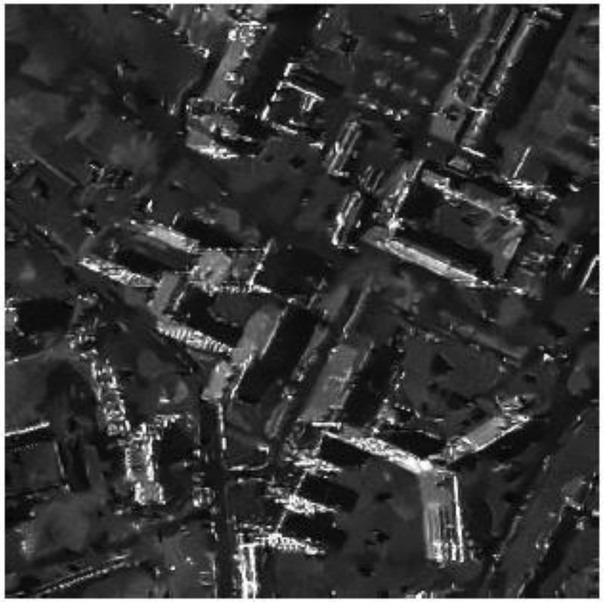
Limits: number of parameters (W, p, number of iterations)

Iterative version

(b) A



(c) $\hat{R}^1$



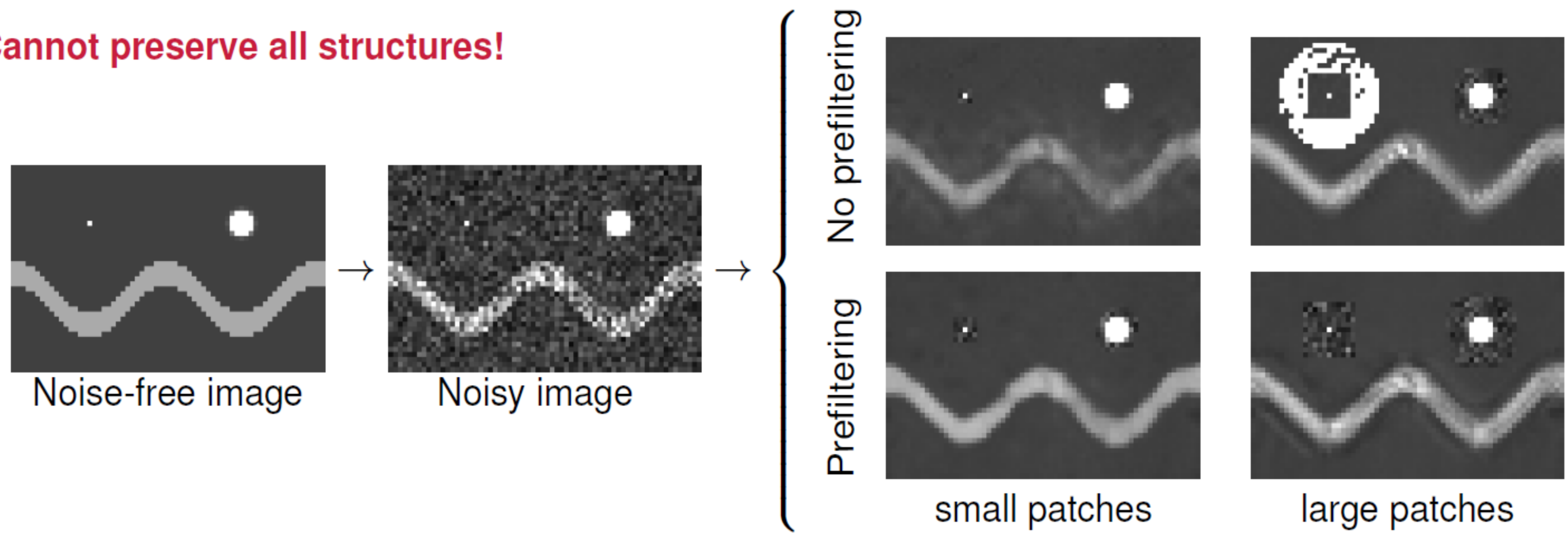
(d) $\hat{R}^i$

- **Introduction**
  - Denoising and models
- **Non-local / patch based approaches**
  - Principle
  - Toy examples
  - Limits and solutions
- **Advanced methods**
  - Noise adaptation
  - Iterative approaches
  - Automatic setting of parameters
  - Shape of patches

Spatially adaptive aggregation

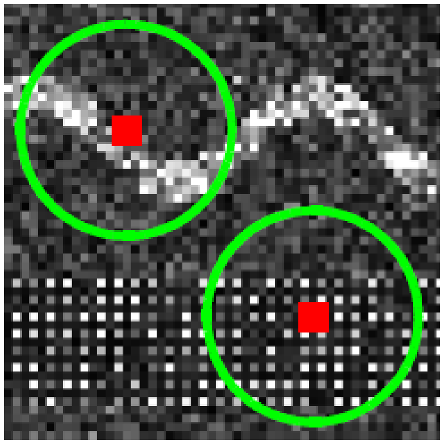
- **Many parameters:**
  - Search window size (rare patch, influence of small weights)
  - Patch size (rare patch effect, noise halo)
  - Number of iterations / pre-filtering strength (bias / variance)
- **Antagonist criteria: no best global tuning**
  - Quality of the estimation / amount of filtering

× **Cannot preserve all structures!**

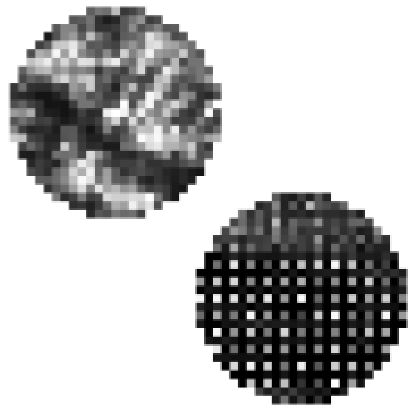


Influence of pre-filtering

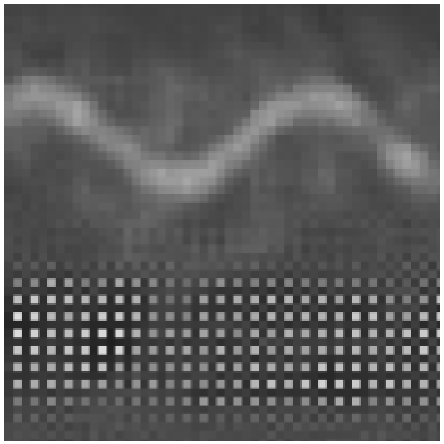
Noisy image



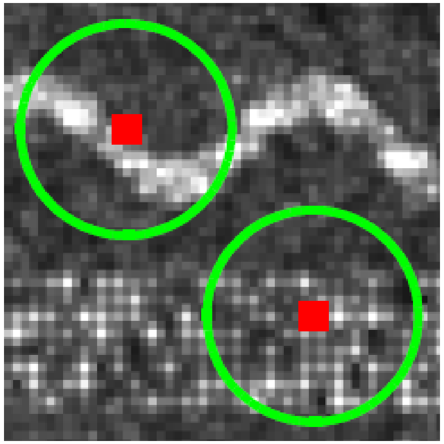
Weights without prefiltering



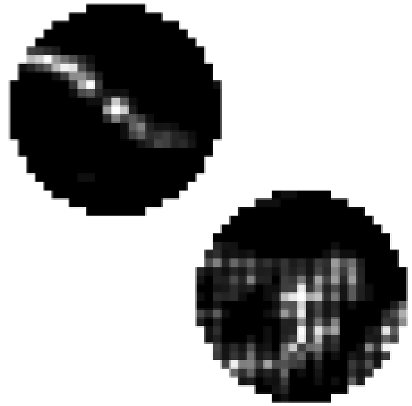
Result without prefiletring



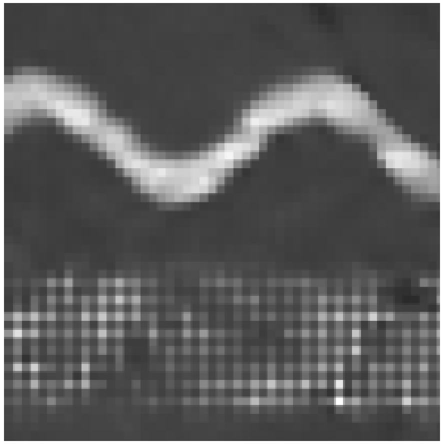
Prefiletred image



Weights with prefiltering



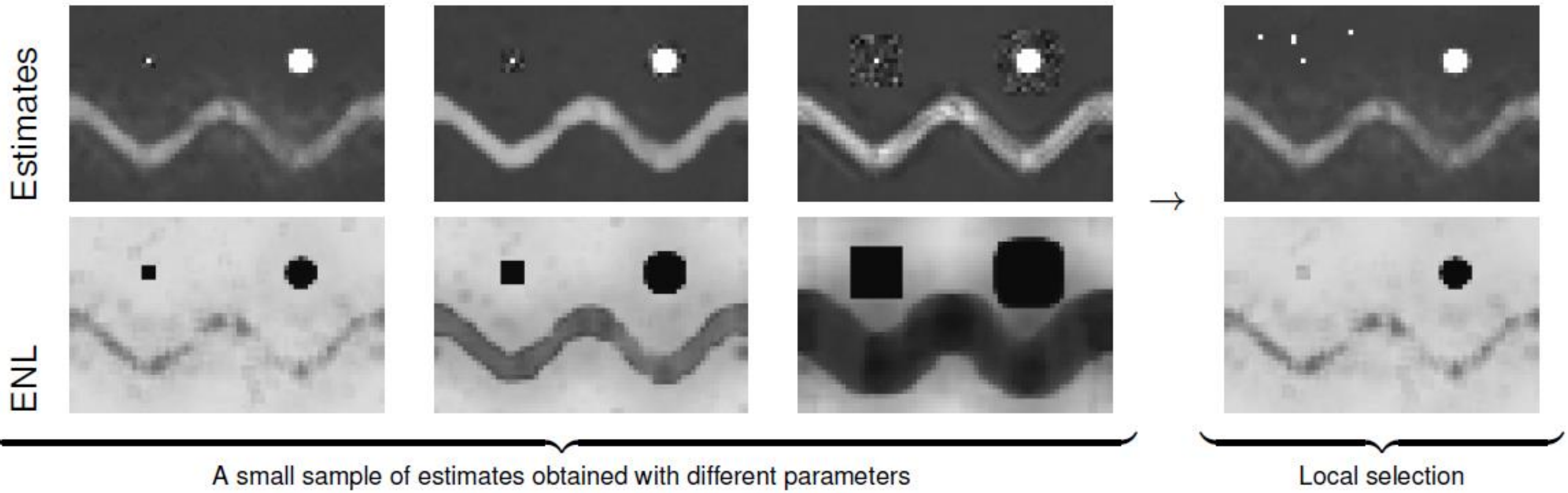
Result with prefiletring



Spatially adaptative aggregation

- **Aggregation:**
  - Compute several estimates with different parameters
  - Select the estimate with the best smoothing

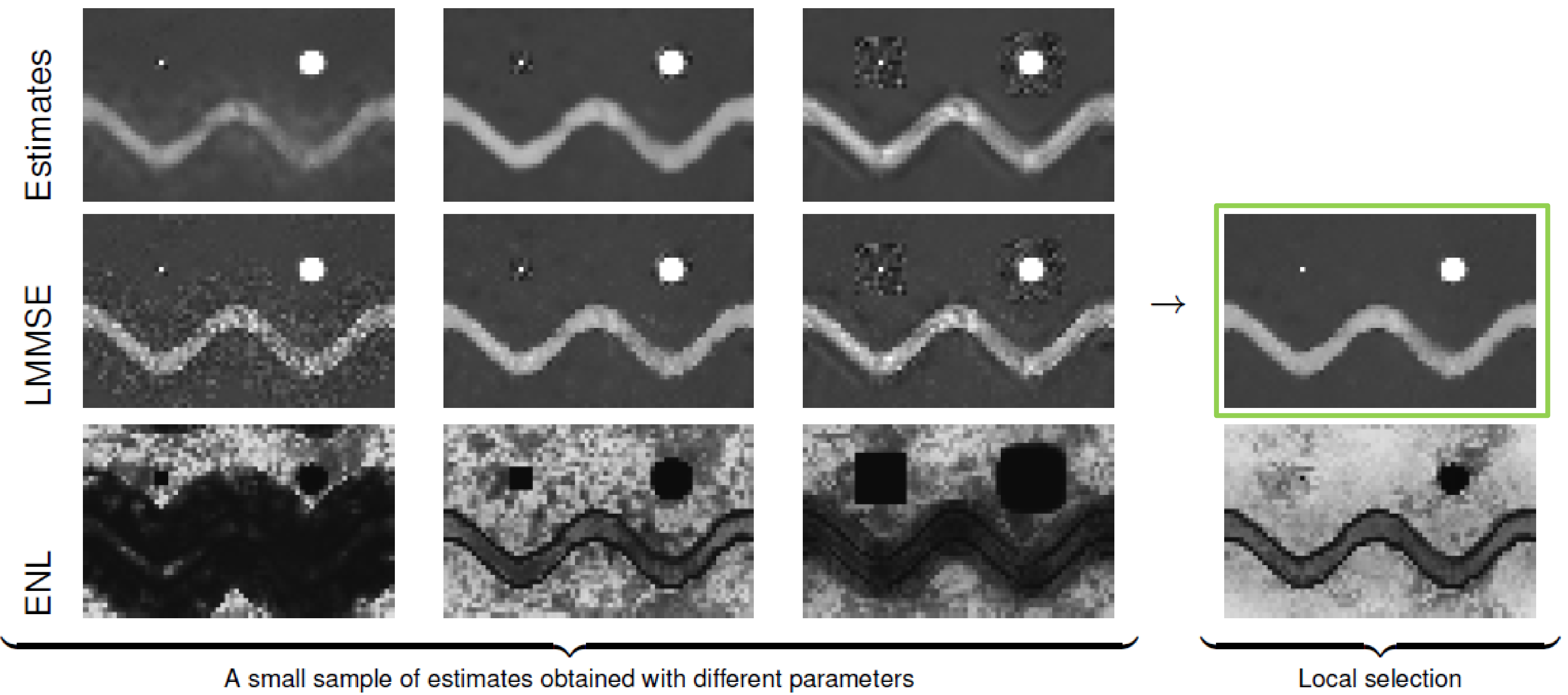
$$\hat{L}^{\mathrm{NL}}(x) = \frac{\left(\sum_{x'} w(x,x')\right)^2}{\sum_{x'} w(x,x')^2}$$



Strong blurring: only takes into account estimation variance but not the bias

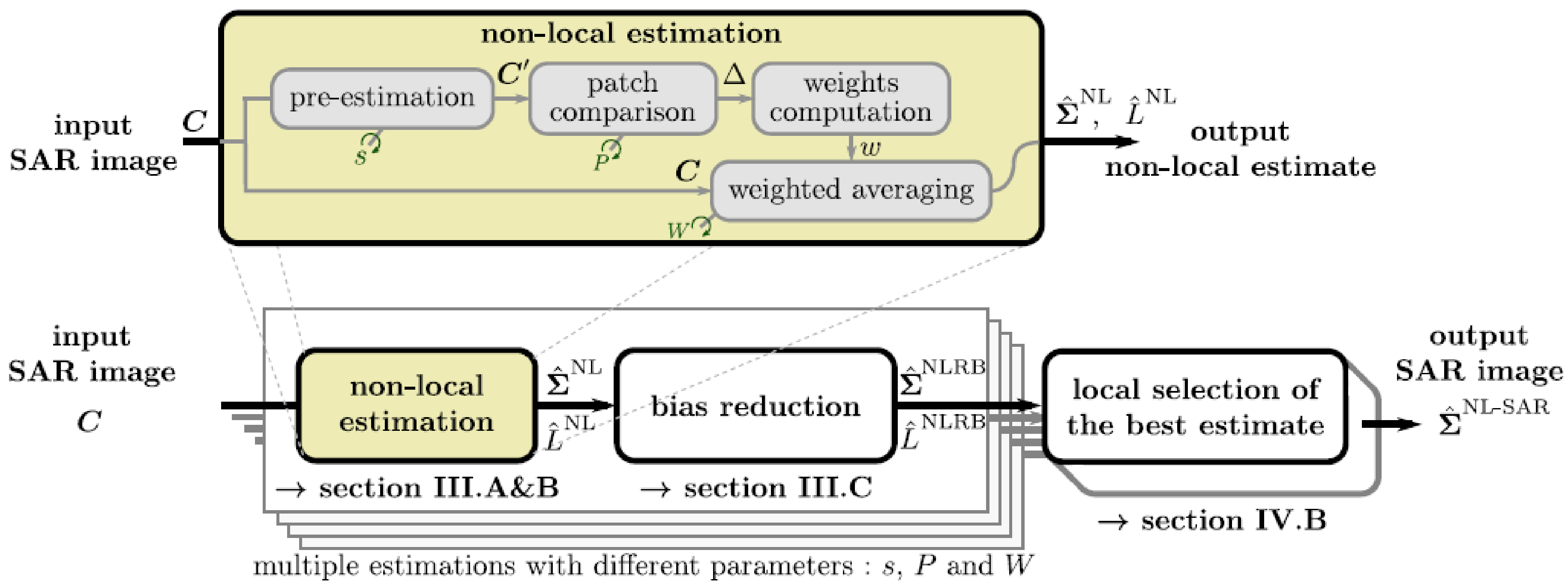
Spatially adaptive aggregation

- **Before aggregation:**
  - Apply bias reduction for each estimation
  - Select the bias reduced estimate with the best smoothing

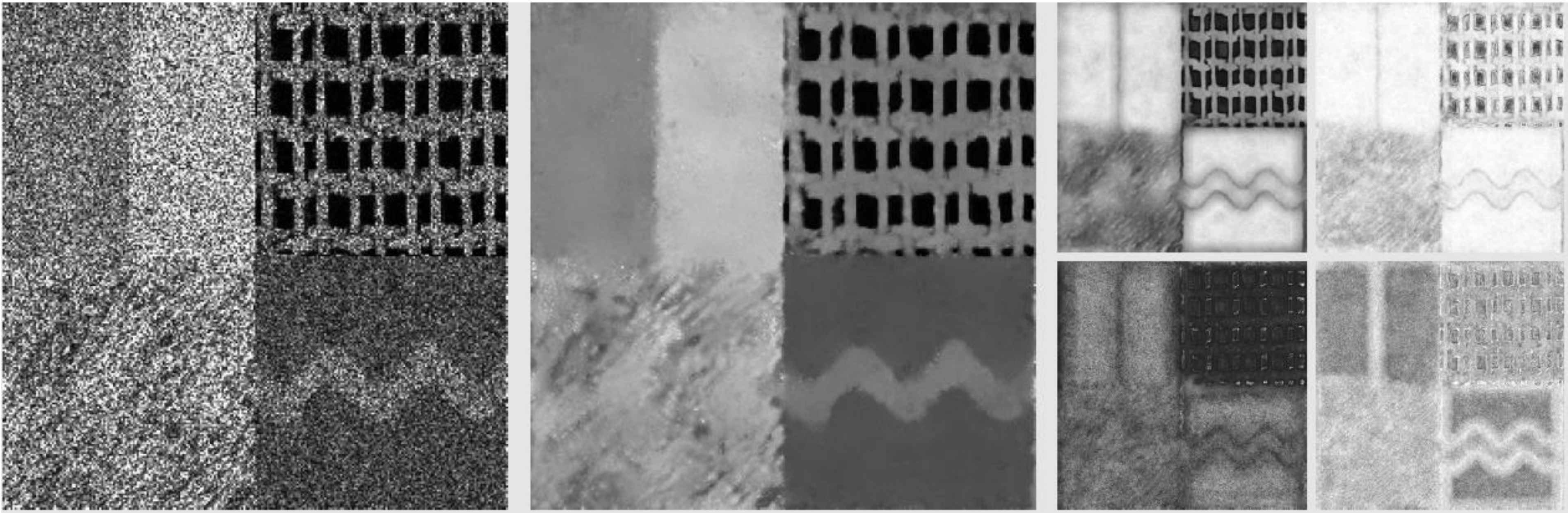


Spatially adaptive aggregation

- **General scheme:**



Example of spatially adaptive aggregation



(a) (b) (c)

(a) Noisy image.

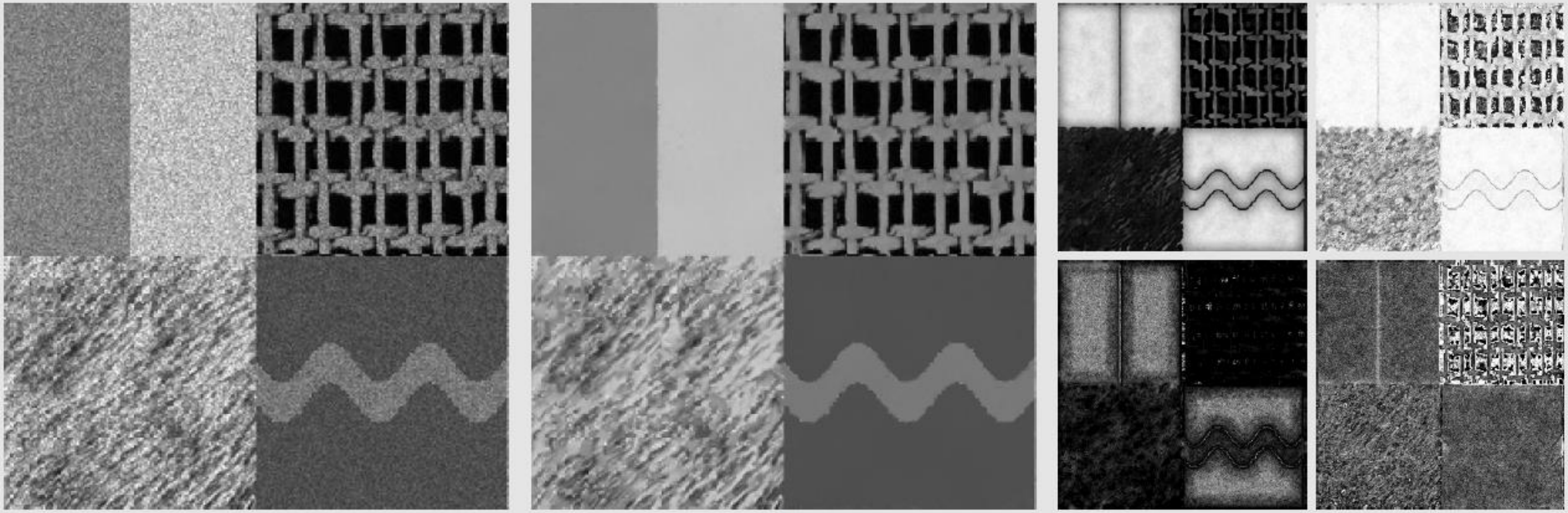
(b) Result of the adaptive approach.

(c) From left to right, top to bottom:

- smoothing strength (range: $[0, 20 \times 20]$),
- search window sizes (range: $[0, 20 \times 20]$),
- the patch size (range: $[3 \times 3, 11 \times 11]$),
- prefiltering strength (range: $[1, 3]$).

NL-SAR: A unified non-local framework for resolution preserving (Pol)(In)SAR, Deledalle, Denis, Tupin, Reigber, Jäger, Pre-print HAL

Example of spatially adaptive aggregation



(a) (b) (c)

(a) Noisy image.

(b) Result of the adaptive approach.

(c) From left to right, top to bottom:

- smoothing strength (range: $[0, 20 \times 20]$),
- search window sizes (range: $[0, 20 \times 20]$),
- the patch size (range: $[3 \times 3, 11 \times 11]$),
- prefiltering strength (range: $[1, 3]$).

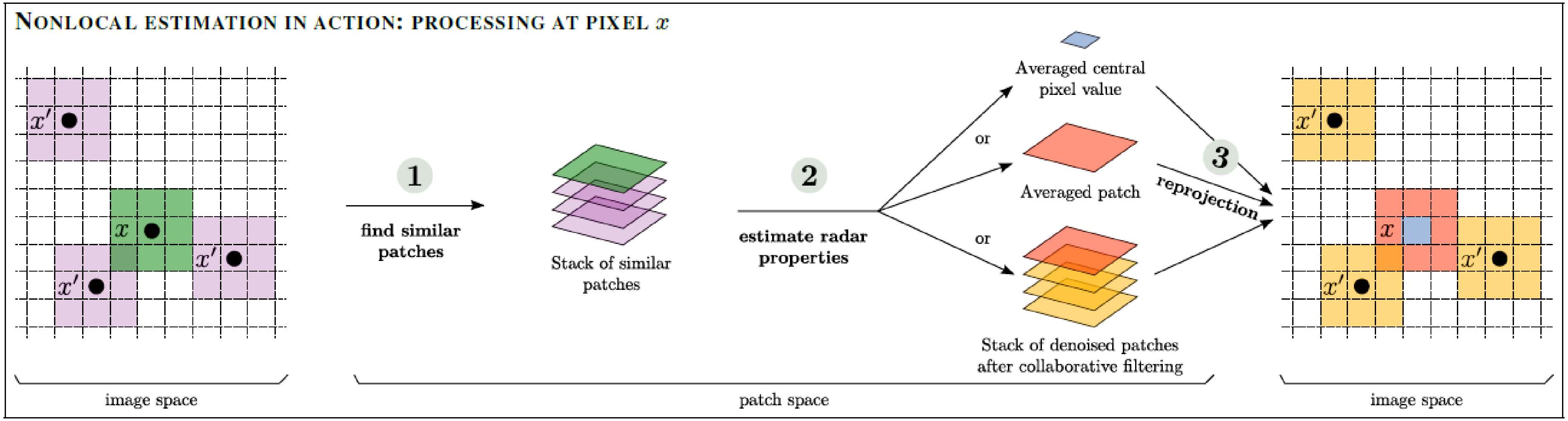
NL-SAR: A unified non-local framework for resolution preserving (Pol)(In)SAR, Deledalle, Denis, Tupin, Reigber, Jäger, Pre-printHAL

Issues

- **Kernel choice**
  - Gaussian is limited (no clear cut)
  - Trapezoïdal kernels

- **Patch shape**
  - Adapted shape
  - Choose the best estimate… by aggregation!

Steps of non-local denoising



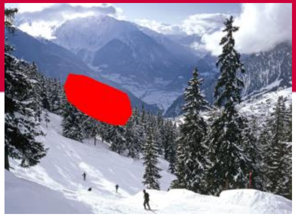
Variations on non-local approaches

- **BM3D**
  - Instead of denoising one pixel: denoise the whole stack of similar patches
- **NL-Bayes**
  - Introduce a prior on the denoised patches (instaed of a ML estimate compute a MAP estimate)
- **Patch dictionnaries**
  - K-SVD
  - Epitomes
  - FoE

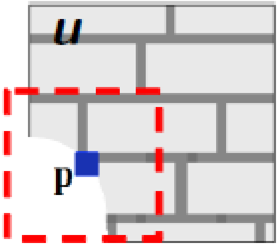
Patch-based applications

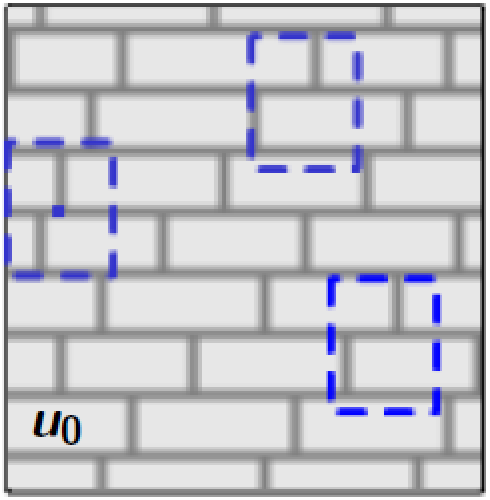
- **Some applications in image processing**
  - Inpainting (image and video)
  - HDR (High Dynamic Range)
  - Texture synthesis

Patch-based inpainting

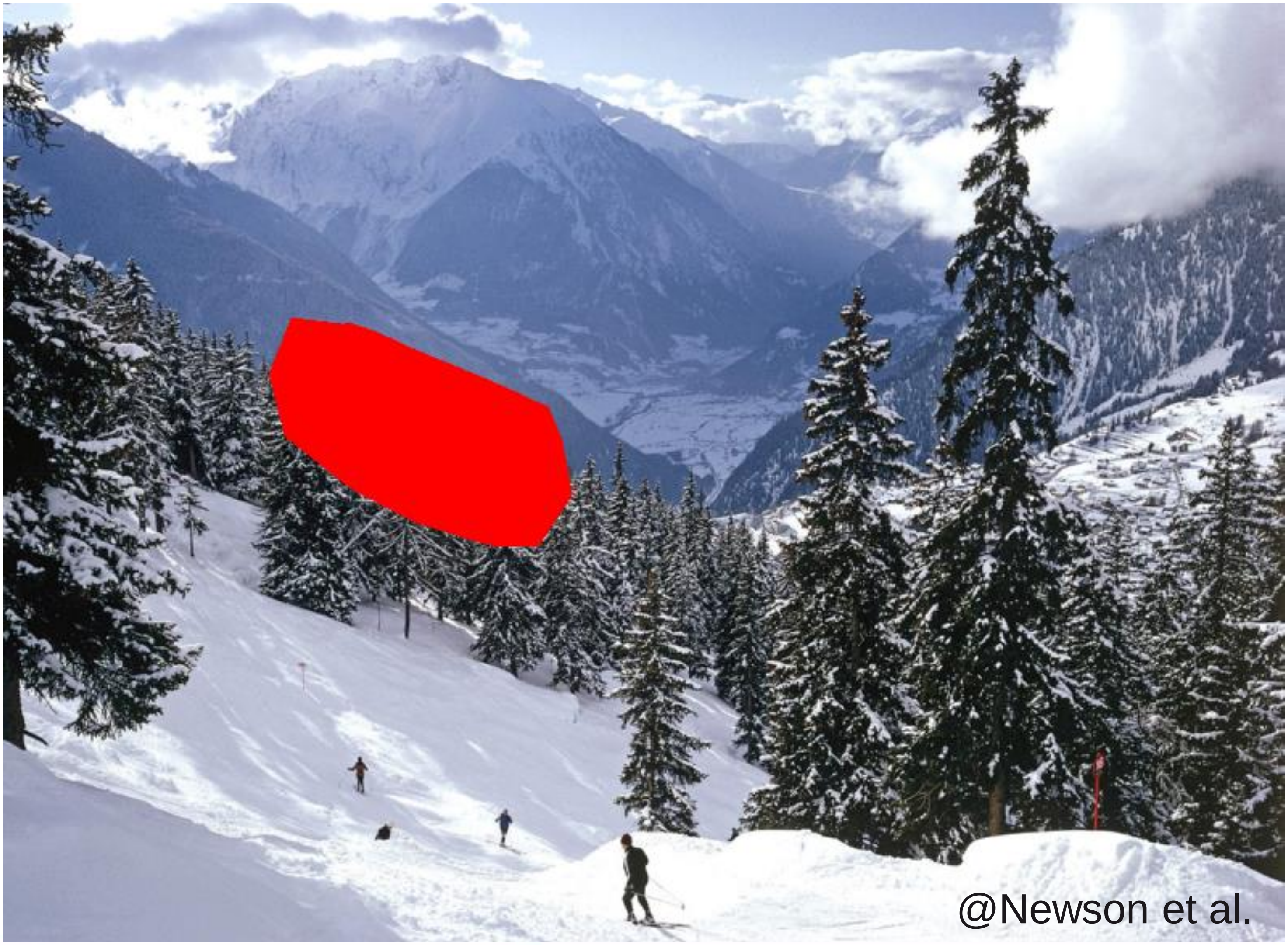


- **Principle:**
  - Start by the boundary pixels of the region to fill
  - Select a patch around the pixels
  - Search for similar patch in the known image
  - Fill the central pixel with the central value

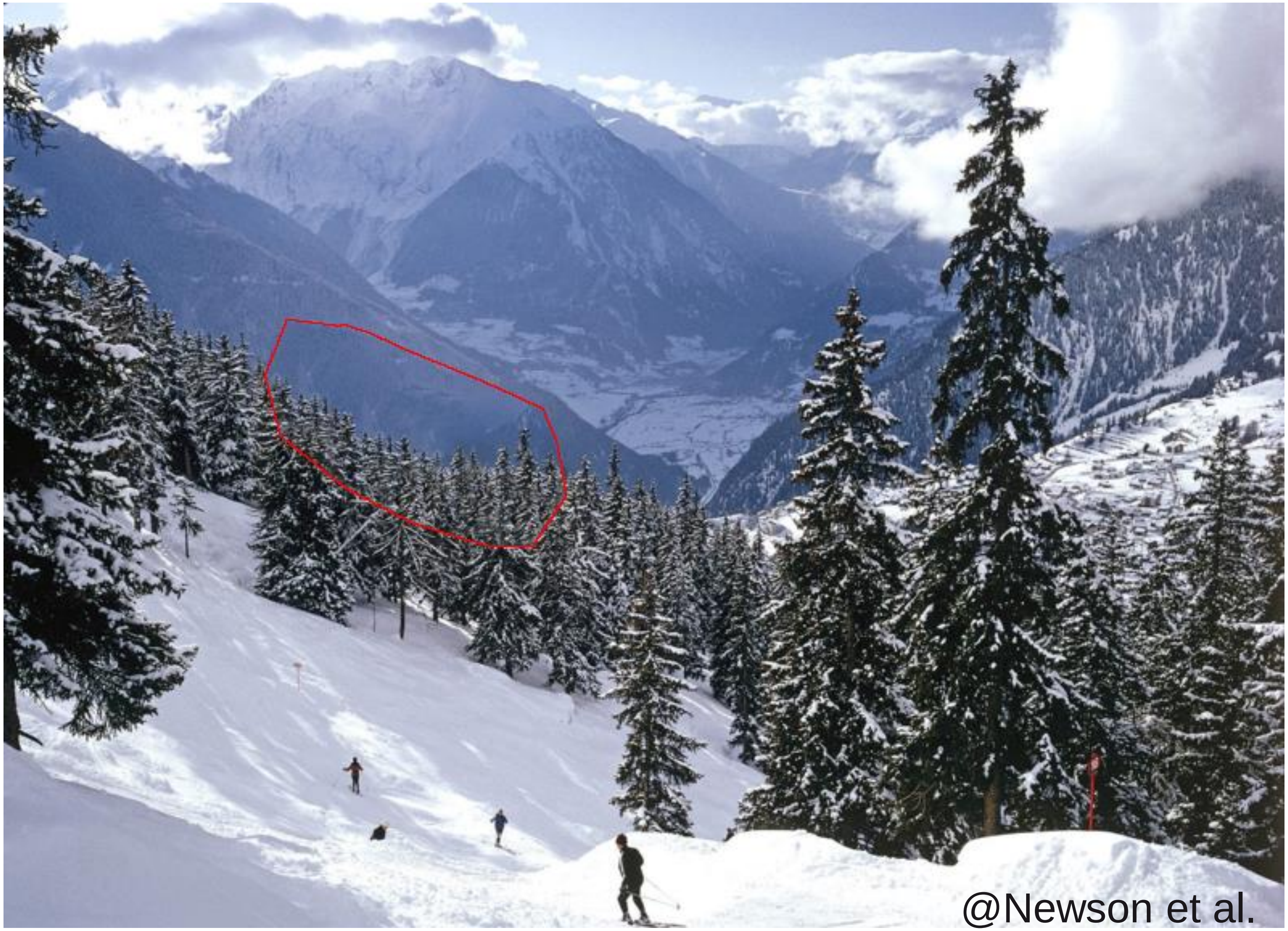




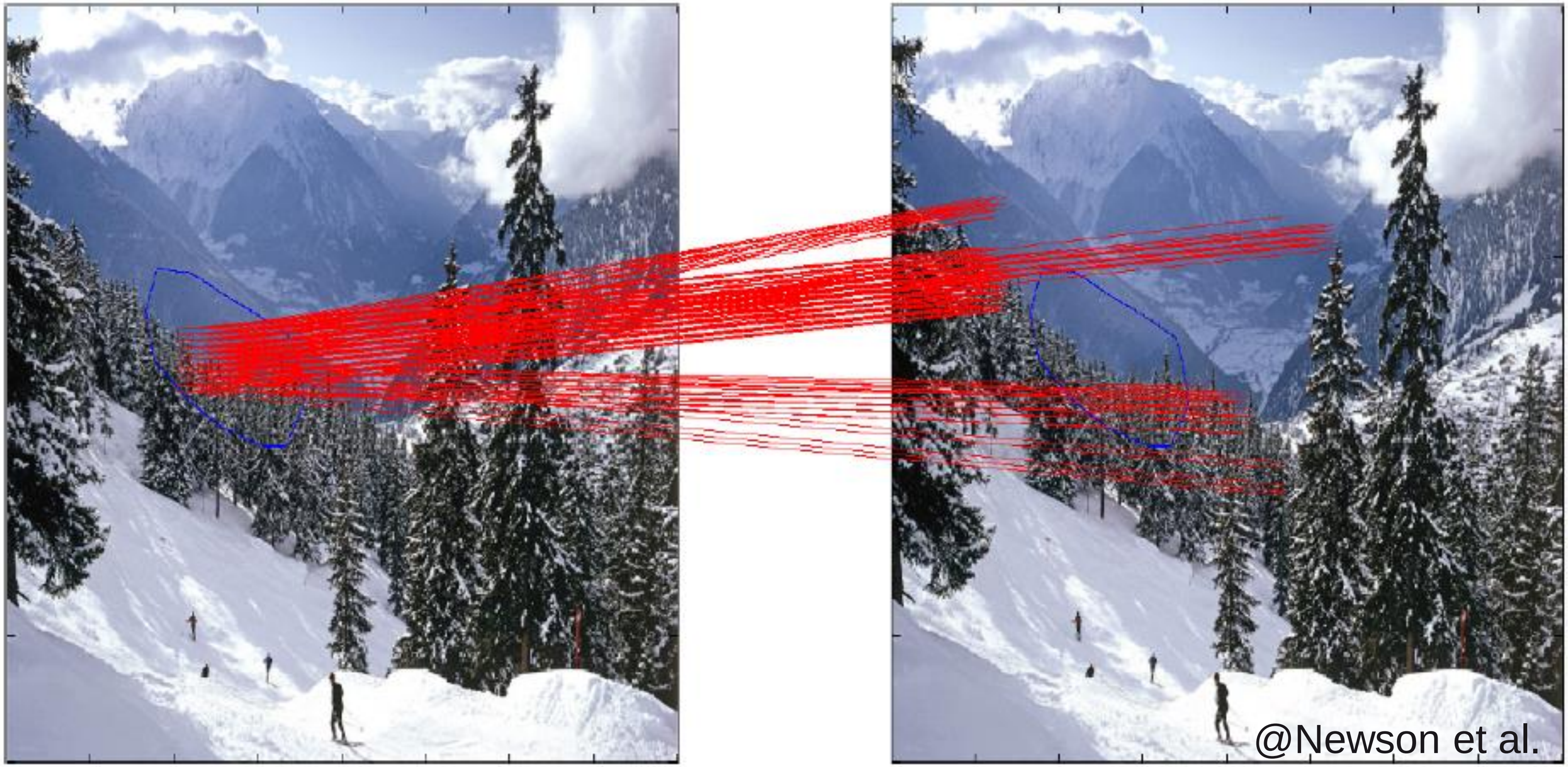
Patch-based inpainting



Patch-based inpainting



Patch-based inpainting



TELECOM ParisTech

Video inpainting

- **Principle**
  - Use space + time patches to fill gaps
  - Multi-resolution framework
  - Estimation of the dominant motion in the video

@Newson et al.

High Dynamic Range Imaging

@Aguerreberre et al.

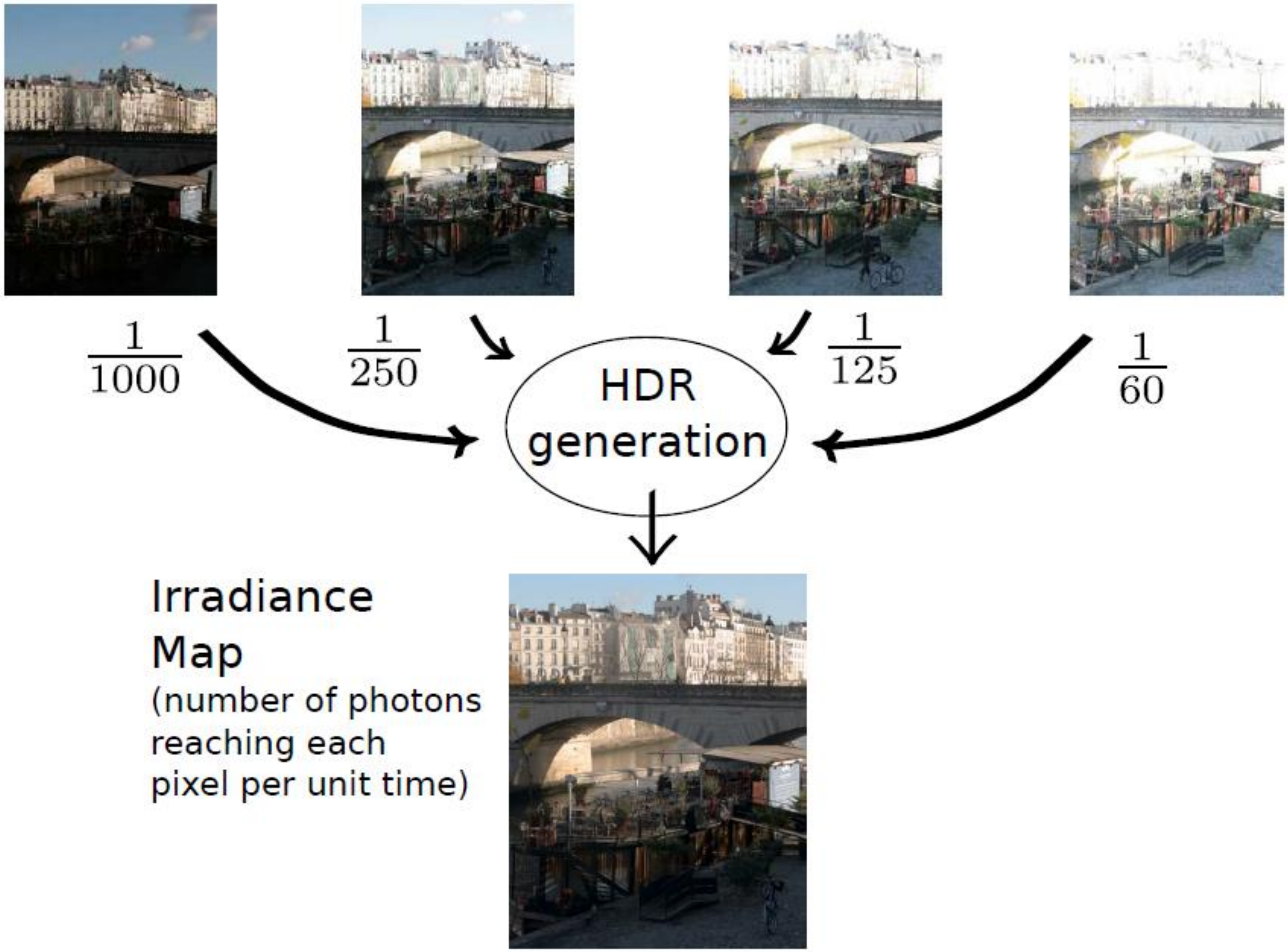
HDR

Loss of details in dark areas

Loss of details in bright areas

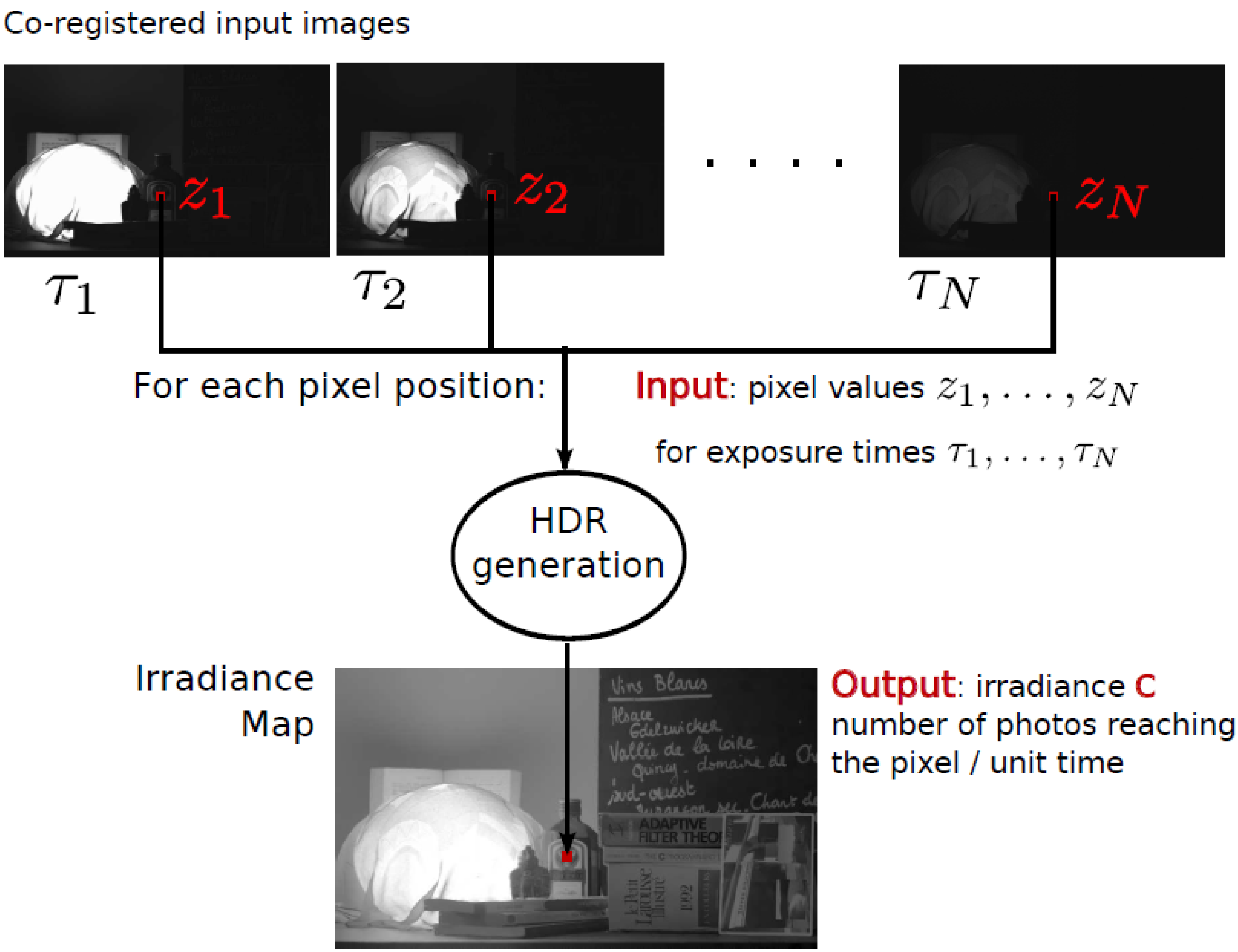
@Aguerreberre et al.

Patch-based HDR (High Dynamic Range)



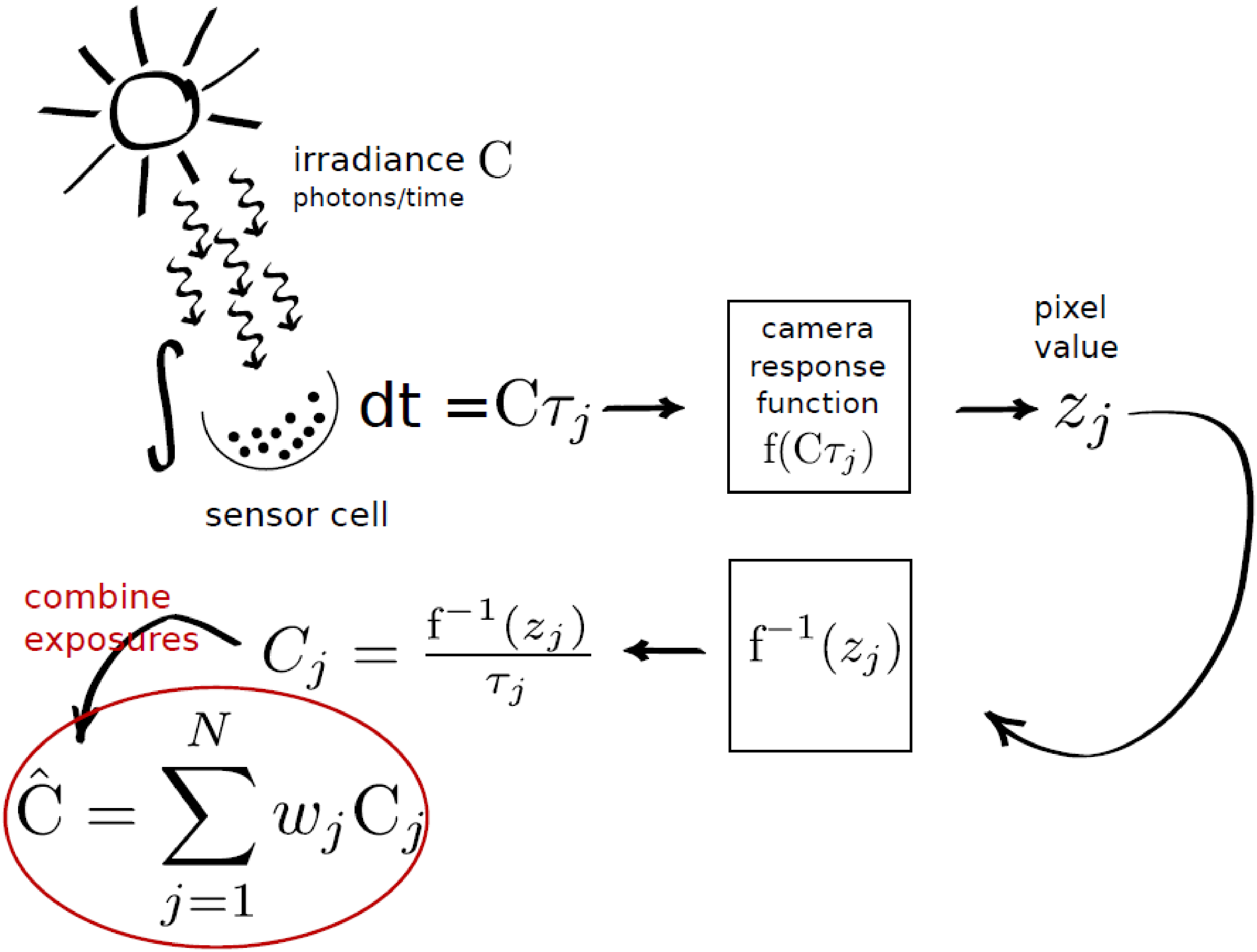
@Aguerreberre et al.

HDR principle – static case



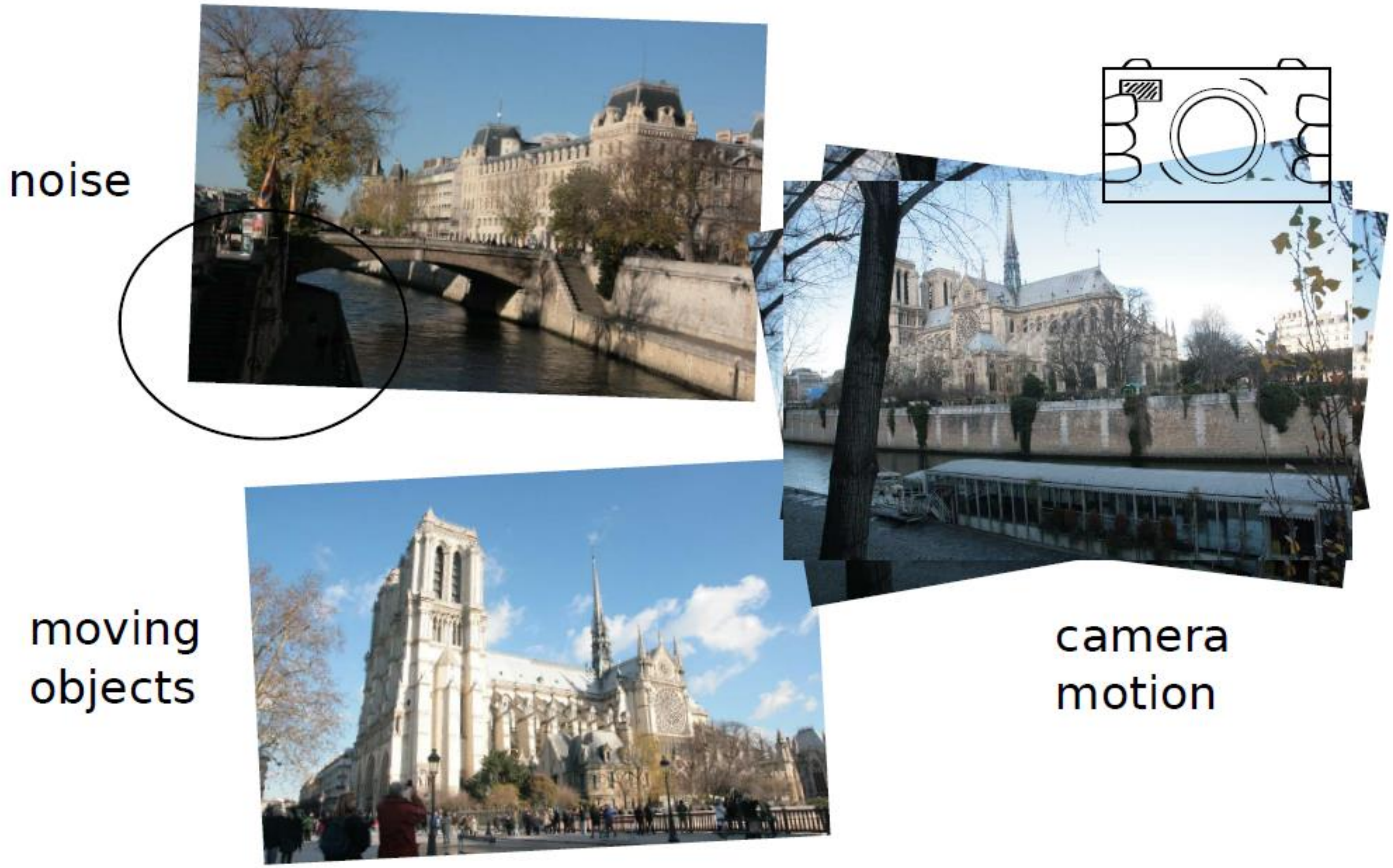
@Aguerreberre et al.

HDR principle – static case



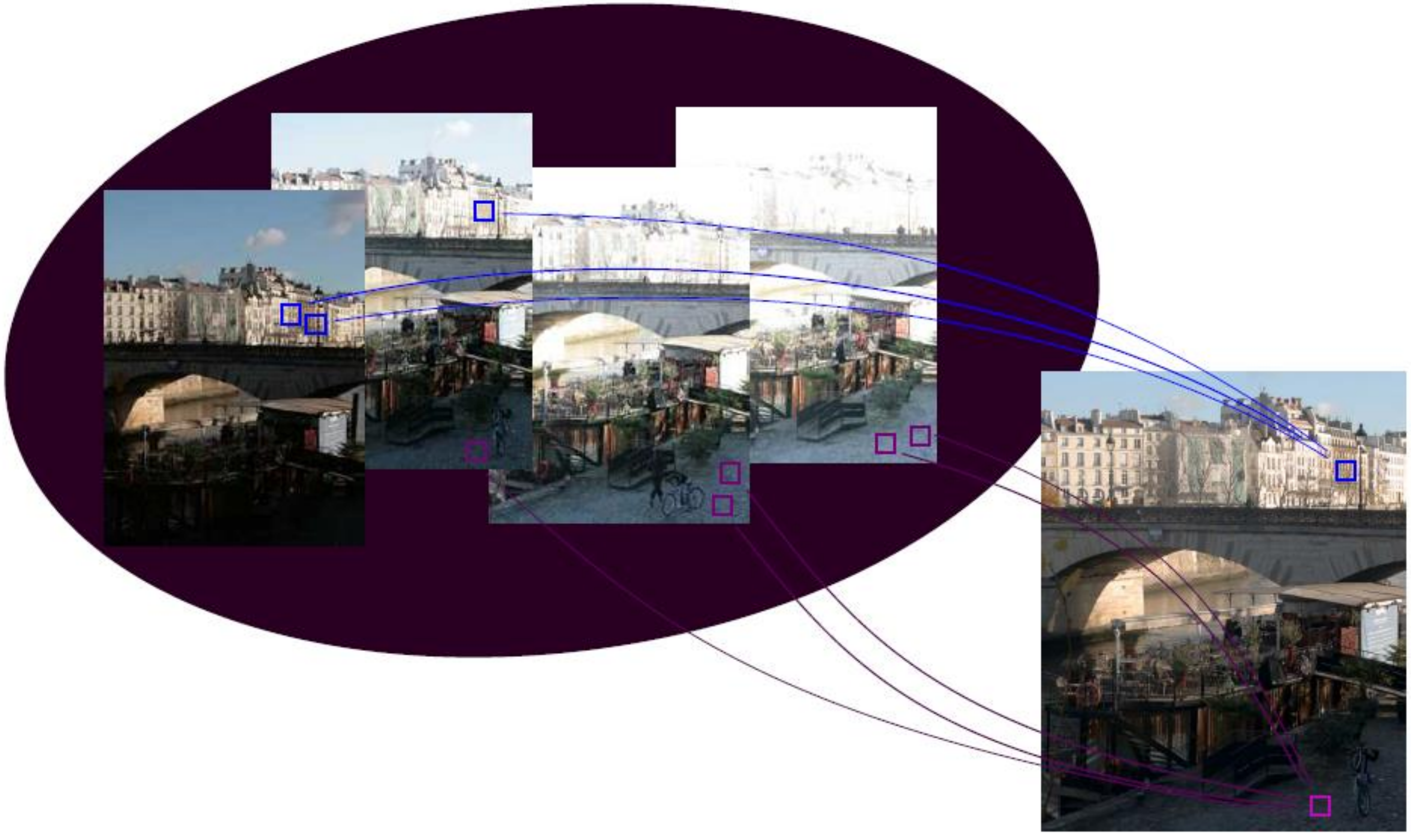
@Aguerreberre et al.

HDR – dynamic case



@Aguerreberre et al.

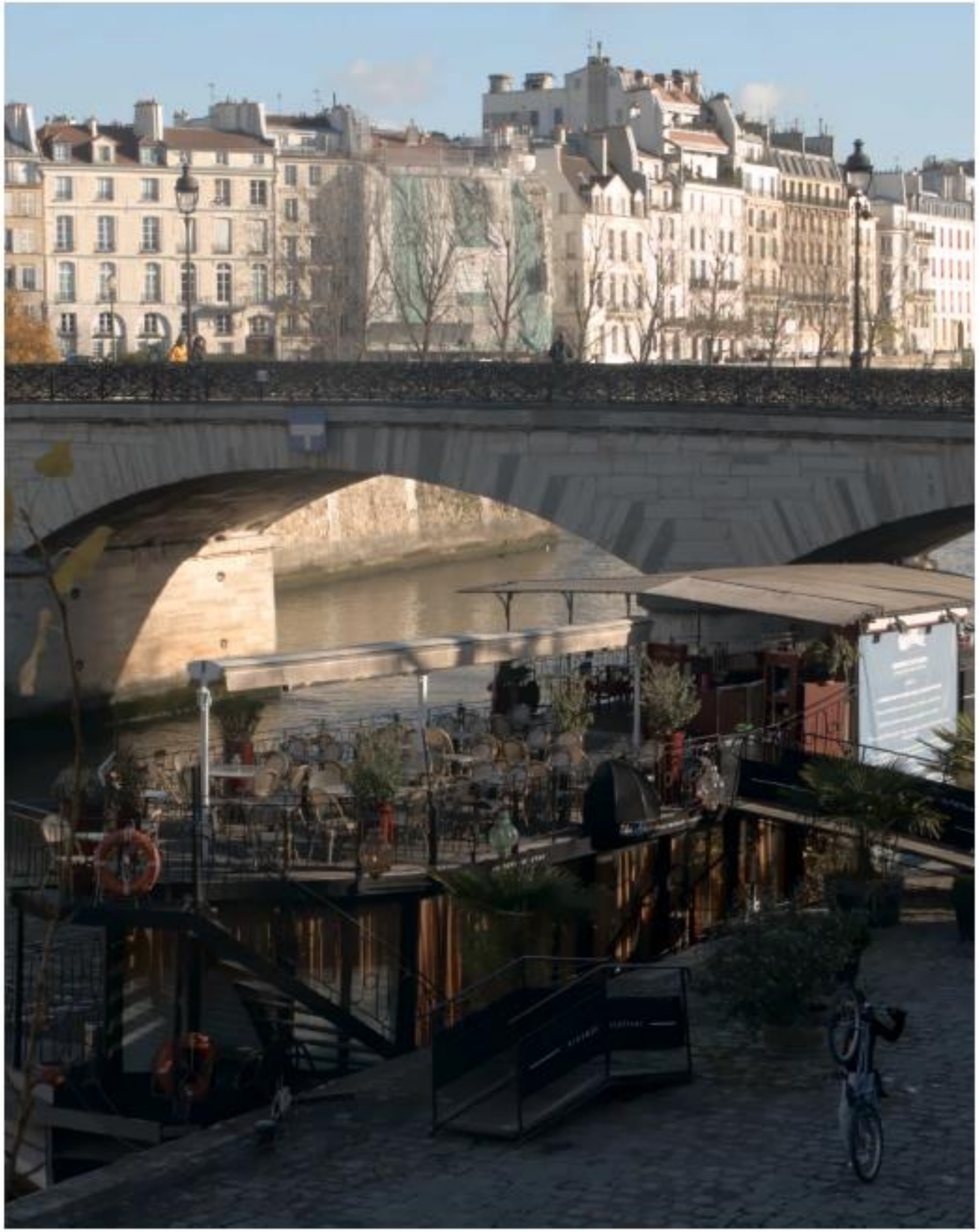
Patch-based HDR



C. Aguerrebere, J. Delon, Y. Gousseau, and P. Musé. Simultaneous HDR image reconstruction and denoising for dynamic scenes. International Conference on Computational Photography (ICCP) 2013.

@Aguerreberre et al.

@Aguerreberre et al.



@Aguerreberre et al.

HDR

Input images

@C. Aguerreberre et al.

@Aguerreberre et al.

HDR

@C. Aguerreberre et al.

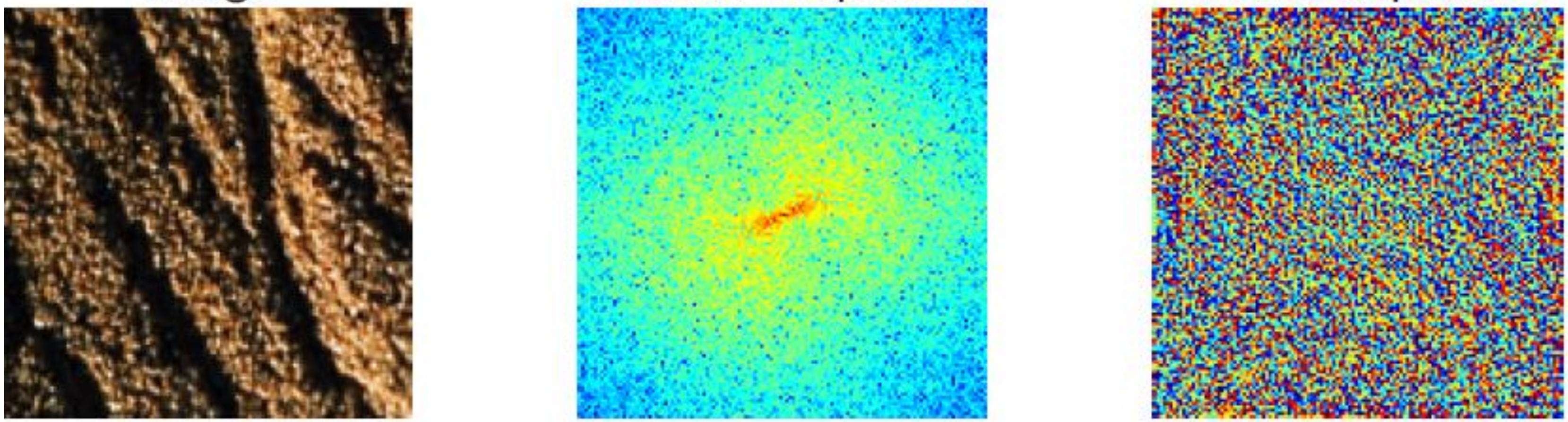
@Aguerreberre et al.

Texture synthesis

@Tartavel et al.

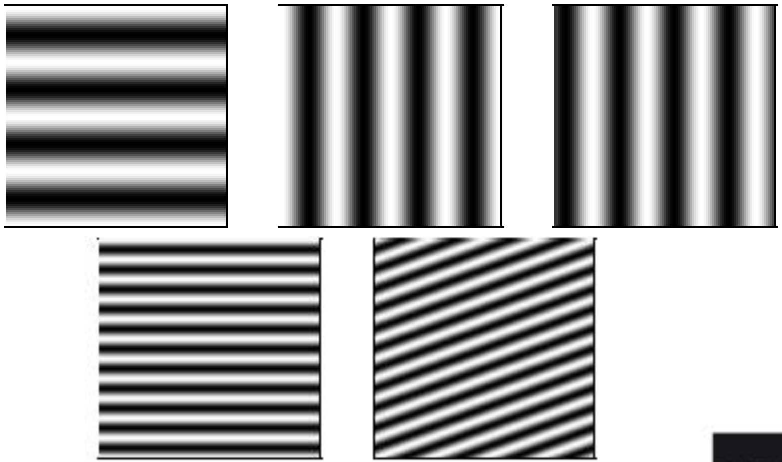
Texture synthesis

Image | Fourier, amplitude | Fourier, phase



Fourier coefficients ⇔ oscillations

- Distance to center: frequency
- Location: orientation
- Phase: position (translation)



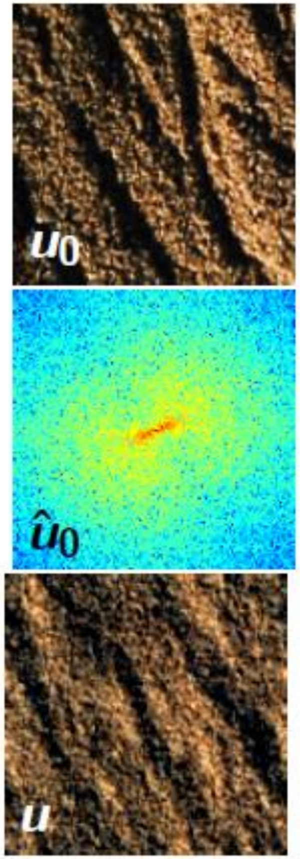
@Tartavel et al.

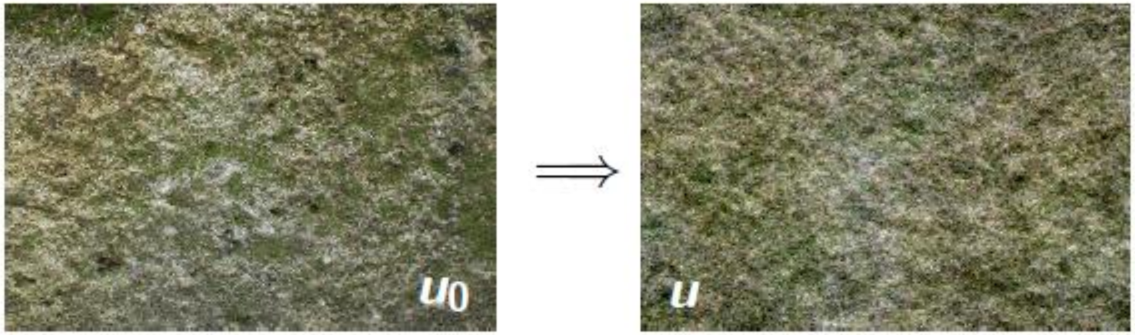
Texture synthesis algorithm:

1. Start from an image u_0
2. Compute its Fourier transform $\hat{u}_0$
3. Generate random phases:

$$\hat{u}(p,q) = \hat{u}_0(p,q) \cdot e^{i\theta(p,q)}$$

4. Compute inverse Fourier transform u from $\hat{u}$





@Tartavel et al.

Texture synthesis – random phase

@Tartavel et al.

Textures synthesis – random phase

Samples | *RPN*


Tiles from roof 1

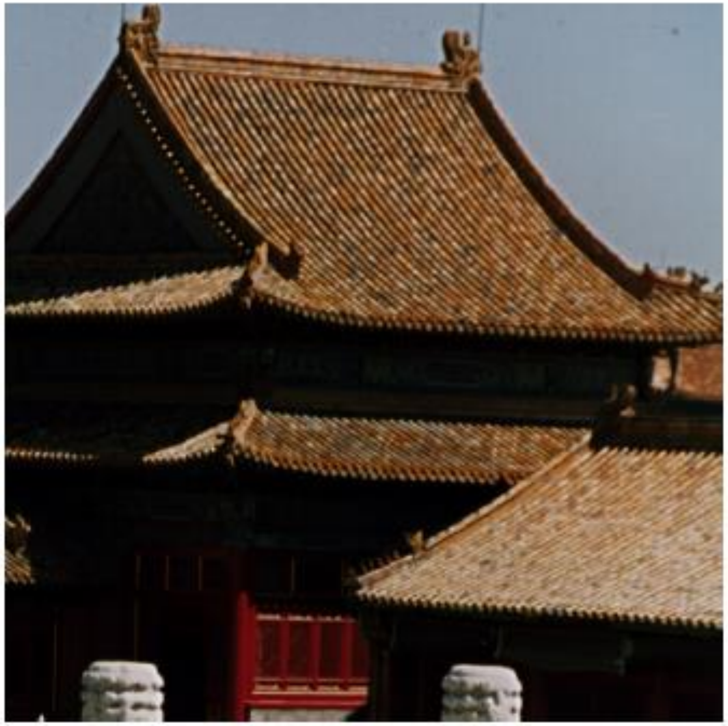

Tiles from roof 1 emulated by RPN


Tiles from roof 2


Tiles from roof 2 emulated by RPN


A macro-texture: tiles at short range


RPN failure to simulate a macro-texture



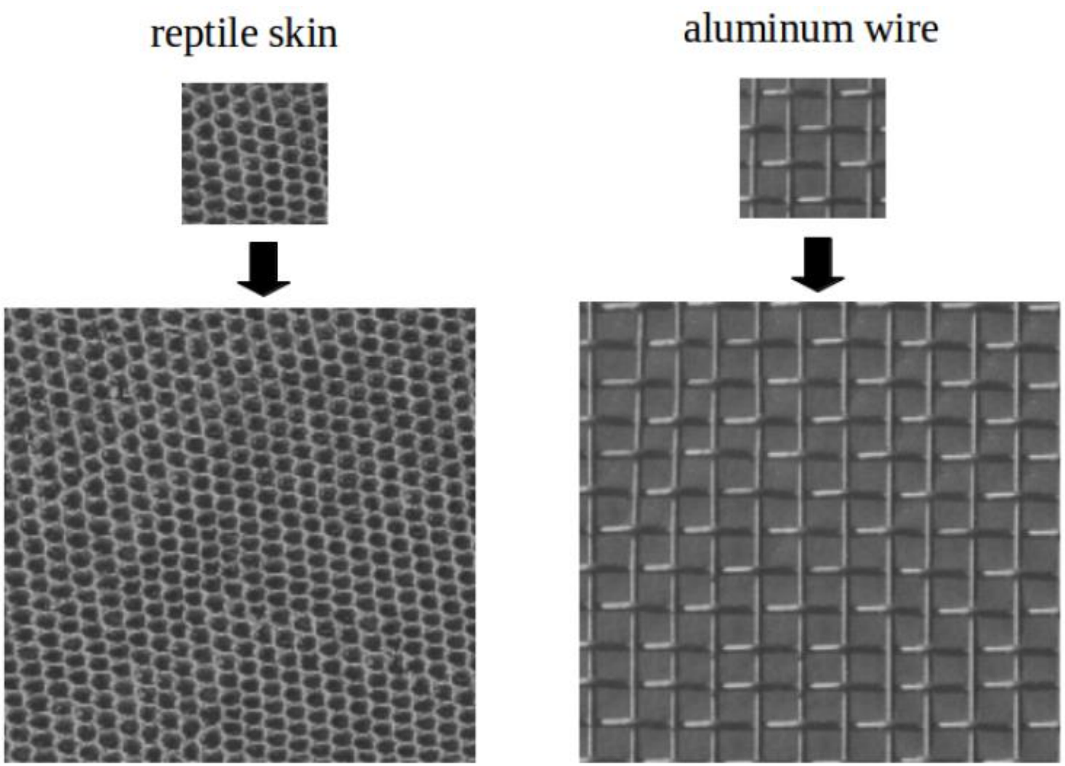
@Tartavel et al.

Textures synthesis – random phase

Input	Synthesis
	
	

@Tartavel et al.

Patch-based synhesis



Synthesis algorithm of *Efros & Leunge*

- Synthesize pixels one by one
- To generate a pixel p:
  1. get the patch around it
  2. find a similar patch in u_0
  3. copy the value of its center pixel

@Tartavel et al.

Synthesis with spectrum and patches

Ideas

- Synthesis from an image u_0
- Mix **Fourier** and **patches**
- Variational approach

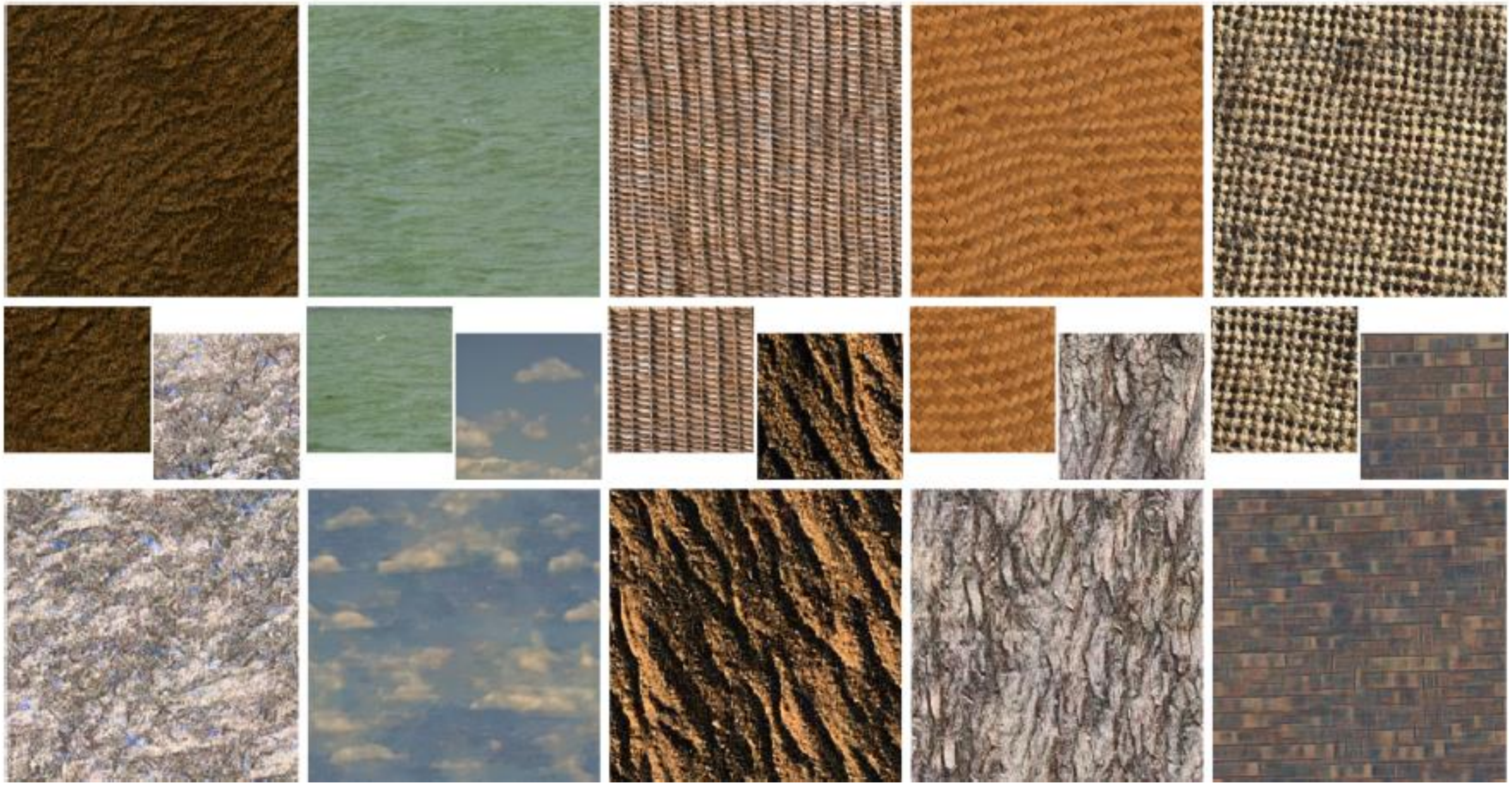
Variational approach

- We define a "similarity" function

$$E_0(u) = \underbrace{S(u, u_0)}_{\text{spectrum}} + \underbrace{P(u, u_0)}_{\text{patches}} + \underbrace{H(u, u_0)}_{\text{histogram}}$$

- We minimize E_0 using the gradient descent algorithm
  $\rightarrow$ local minima are synthesis!

Examples



@Tartavel et al.

Examples

@Tartavel et al.

Conclusion

- **Patch-based approach for image processing**
  - Very powerful and « weak » models
  - General formulation
  - Wide range of applications beyond denoising
  - Spatial and temporal adaptation (video)

- **Limits**
  - Additive gaussian noise
  - Many parameters

Acknowledgments / References

- **PhD students :**
  - C. Deledalle, V. Duval, C. Aguerrebere, A. Newson, G. Tartavel
- **Publications :**
  - Video inpainting of complex scenes, A. Newson et al., SIAM 2014
  - Simultaneous HDR reconstruction and denoising of dynamic scenes, C. Aguerrebere et al., ICCP 2013
  - A probabilistic patch based approach, C. Deledalle et al., IEEE IP 2009
  - Variational texture synthesis with sparcity and spectrum constraints, G. Tartavel et al., 2015