A brief survey of patch based method

E. Le Pennec and J. Salmon

SELECT - INRIA Saclay IdF, Université Paris Sud (Orsay)

April 2011 - Santa Fe

Patch based method

Framework

- Estimation of an image I from a noisy observation Y
$$Y = I + \sigma\mathcal{E} \qquad (\mathcal{E} \text{ Gaussian white noise})$$
- Regression, Texture synthesis, Image completion...

Patch based method

Framework

- Estimation of an image I from a noisy observation Y
  $$Y = I + \sigma\mathcal{E} \qquad (\mathcal{E} \text{ Gaussian white noise})$$
- Regression, Texture synthesis, Image completion...

Image, pixel and patch

- Patch = Neighborhood of a pixel.
- Image $\Leftrightarrow$ { pixel } $\rightarrow$ estimation of pixels.
- Image $\Leftrightarrow$ { patch } $\rightarrow$ estimation of patches + reprojection.

Patch based method

Framework

- Estimation of an image I from a noisy observation Y
$$Y = I + \sigma\mathcal{E} \qquad (\mathcal{E} \text{ Gaussian white noise})$$
- Regression, Texture synthesis, Image completion...

Image, pixel and patch

- Patch = Neighborhood of a pixel.
- Image $\Leftrightarrow$ { pixel } $\rightarrow$ estimation of pixels.
- Image $\Leftrightarrow$ { patch } $\rightarrow$ estimation of patches + reprojection.

Three-Step Methods

- Step 1: Construct a set of noisy patches.
- Step 2: Estimate those patches.
- Step 3: Combine those patch estimates at the pixel level.

Patch based method

Framework

- Estimation of an image I from a noisy observation Y
  $$Y = I + \sigma\mathcal{E} \qquad (\mathcal{E} \text{ Gaussian white noise})$$
- Regression, Texture synthesis, Image completion...

Image, pixel and patch

- Patch = Neighborhood of a pixel.
- Image $\Leftrightarrow$ { pixel } $\rightarrow$ estimation of pixels.
- Image $\Leftrightarrow$ { patch } $\rightarrow$ estimation of patches + reprojection.

Three-Step Methods

- Step 1: Construct a set of noisy patches.
- Step 2: Estimate those patches.
- Step 3: Combine those patch estimates at the pixel level.

Outline

1. A brief survey of patch based method
   - Pixels and Patches
   - Texture synthesis, image completion and denoising
2. A survey of patch based estimation
   - Toward patch based estimators
   - NL-Means and interpretations
   - BM3D and other state-of-the-art methods
3. Statistical aggregation
   - Initial estimates and aggregation
   - PAC-Bayesian aggregation
4. Patches and Aggregation
   - Framework and theory?
   - Patchwise aggregation
   - Pixelwise aggregation

Outline

Pixels and patches

From pixel value to patch

- Patch $P^I(i)$: small sub-image $P^I(i)[\delta] = I(i + \delta)$ with $\delta \in V_W$.
- Example: square patches with $V_W = \{-W_- \leq \delta_1, \delta_2 \leq W_+\}$.

Pixels and patches

From pixel value to patch

- Patch $P^I(i)$: small sub-image $P^I(i)[\delta] = I(i+\delta)$ with $\delta \in V_W$.
- Example: square patches with $V_W = \{-W_- \leq \delta_1, \delta_2 \leq W_+\}$.

Patch and images

- From $I = (I(i))$ to $P^I = ((P^I(i)[\delta]))$: lifting of a 1-D image to a W^2-D image.
- From P^I to I: *easy* (naive) inverse using $I(i) = P^I(i)[0]$

Pixels and patches

From pixel value to patch

- Patch $P^I(i)$: small sub-image $P^I(i)[\delta] = I(i+\delta)$ with $\delta \in V_W$.
- Example: square patches with $V_W = \{-W_- \leq \delta_1, \delta_2 \leq W_+\}$.

Patch and images

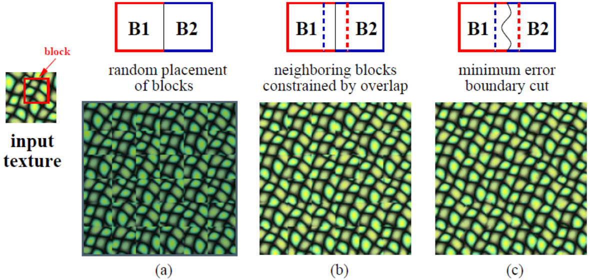
- From $I = (I(i))$ to $P^I = ((P^I(i)[\delta]))$: lifting of a 1-D image to a W^2-D image.
- From P^I to I: *easy* (naive) inverse using $I(i) = P^I(i)[0]$

Use in texture synthesis

Quilting

- Generate a new texture by quilting patch with similar context than the original one.
- Efros and Leung, *Texture Synthesis by Non-parametric Sampling* (1999)
- Efros and Freeman, *Image Quilting for Texture Synthesis and Transfer* (2001)

Use in texture synthesis



Quilting

- Generate a new texture by quilting patch with similar context than the original one.
- Efros and Leung, *Texture Synthesis by Non-parametric Sampling* (1999)
- Efros and Freeman, *Image Quilting for Texture Synthesis and Transfer* (2001)

Use in texture synthesis

Quilting

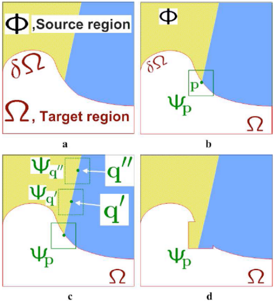
- Generate a new texture by quilting patch with similar context than the original one.
- Efros and Leung, *Texture Synthesis by Non-parametric Sampling* (1999)
- Efros and Freeman, *Image Quilting for Texture Synthesis and Transfer* (2001)

Use in image completion

Inpainting

- Structure propagation by exemplar-based texture synthesis.
- Criminisi, Pérez and Toyama, *Region Filling and Object Removal by Exemplar-Based Image Inpainting* (2004)

Use in image completion



Inpainting

- Structure propagation by exemplar-based texture synthesis.
- Criminisi, Pérez and Toyama, *Region Filling and Object Removal by Exemplar-Based Image Inpainting* (2004)

Use in image completion

Inpainting

- Structure propagation by exemplar-based texture synthesis.
- Criminisi, Pérez and Toyama, *Region Filling and Object Removal by Exemplar-Based Image Inpainting* (2004)

Use in denoising

NL-Means

- Average patches that are similar.
- Buades, Coll and Morel, *A review of image denoising algorithms, with a new one* (2005)

Use in denoising

NL-Means

- Average patches that are similar.
- Buades, Coll and Morel, *A review of image denoising algorithms, with a new one* (2005)
- Much more than that!

Use in denoising

NL-Means

- Average patches that are similar.
- Buades, Coll and Morel, *A review of image denoising algorithms, with a new one* (2005)
- Much more than that!

Outline

Images, noise and estimate

Image $N \times N$

- $I(i) = I(i_1, i_2) \in \mathbb{R}$ with $i = (i_1, i_2) \in [1, N]^2$.
- Loss: L_2 norm (quadratic loss)

Images, noise and estimate

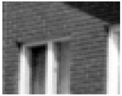


Image $N \times N$

- $I(i) = I(i_1, i_2) \in \mathbb{R}$ with $i = (i_1, i_2) \in [1, N]^2$.
- Loss: L_2 norm (quadratic loss)

Noisy observation

- $Y(i) = I(i) + \sigma\mathcal{E}(i)$.
- $\mathcal{E}$ i.i.d. standard Gaussian noise with known variance σ^2.
- Other noises possible (bounded,...)

Images, noise and estimate

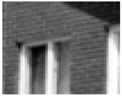
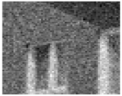
 

Image $N \times N$

- $I(i) = I(i_1, i_2) \in \mathbb{R}$ with $i = (i_1, i_2) \in [1, N]^2$.
- Loss: L_2 norm (quadratic loss)

Noisy observation

- $Y(i) = I(i) + \sigma\mathcal{E}(i)$.
- $\mathcal{E}$ i.i.d. standard Gaussian noise with known variance σ^2.
- Other noises possible (bounded,...)

Estimation

- Estimate $I(i)$ by $\widehat{I}(i)$ from whole Y.
- Non local behavior possible...

Images, noise and estimate

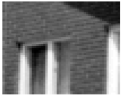
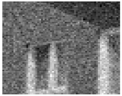
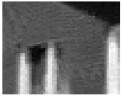
  

Image $N \times N$

- $I(i) = I(i_1, i_2) \in \mathbb{R}$ with $i = (i_1, i_2) \in [1, N]^2$.
- Loss: L_2 norm (quadratic loss)

Noisy observation

- $Y(i) = I(i) + \sigma \mathcal{E}(i)$.
- $\mathcal{E}$ i.i.d. standard Gaussian noise with known variance σ^2.
- Other noises possible (bounded,...)

Estimation

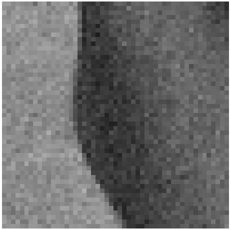
- Estimate $I(i)$ by $\widehat{I}(i)$ from whole Y.
- Non local behavior possible...

Kernel methods

Generic kernel methods

- Estimate $I(i)$ by an average $\widehat{I}(i) = \sum_{k \in [1,N]^2} \theta_{i,k} Y(k)$
- The weights $\theta_{i,k}$ may (and will) depend on i and k as well on Y.

Kernel methods



Target pixel i: center of the sub-image

Generic kernel methods

- Estimate $I(i)$ by an average $\widehat{I}(i) = \sum_{k \in [1,N]^2} \theta_{i,k} Y(k)$
- The weights $\theta_{i,k}$ may (and will) depend on i and k as well on Y.

Some classical filters: $\widehat{I}(i) = \Sigma_k \theta_{i,k} Y(k)$

Classical kernel - Nadaraya (64) , Watson (64)

$$\theta_{i,k} = \frac{K_h(i,k)}{\sum_{k'} K_h(i,k')} \text{ (no dependency on } Y\text{)}$$

- K: kernel and h: window size / smoothing parameter
- Gaussian kernel: $K_h(i,j) = e^{-((i_1-k_1)^2+(i_2-k_2)^2)/2h^2}$

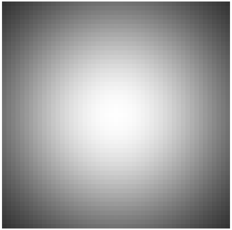
Some classical filters: $\widehat{I}(i) = \Sigma_k \theta_{i,k} Y(k)$

Classical kernel - Nadaraya (64) , Watson (64)

$$\theta_{i,k} = \frac{K_h(i,k)}{\sum_{k'} K_h(i,k')} \text{ (no dependency on } Y\text{)}$$

- K: kernel and h: window size / smoothing parameter
- Gaussian kernel: $K_h(i,j) = e^{-((i_1-k_1)^2+(i_2-k_2)^2)/2h^2}$

Some classical filters: $\widehat{I}(i) = \Sigma_k \theta_{i,k} Y(k)$



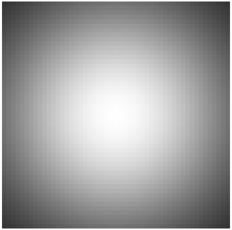
$\theta_{i,k}$: Gaussian

Classical kernel - Nadaraya (64) , Watson (64)

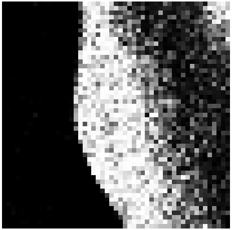
$$\theta_{i,k} = \frac{K_h(i,k)}{\sum_{k'} K_h(i,k')} \text{ (no dependency on } Y\text{)}$$

- K: kernel and h: window size / smoothing parameter
- Gaussian kernel: $K_h(i,j) = e^{-((i_1-k_1)^2+(i_2-k_2)^2)/2h^2}$

Some classical filters: $\widehat{I}(i) = \Sigma_k \theta_{i,k} Y(k)$



$\theta_{i,k}$: Gaussian



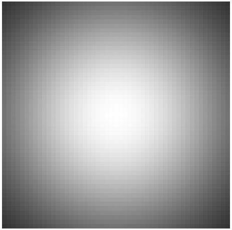
$\theta_{i,k}$: Yaroslavsky

Yaroslavsky's filter - Yaroslavsky (85), Lee (83)

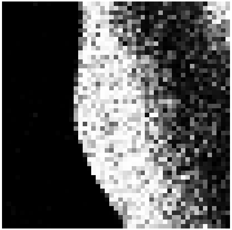
$$\theta_{i,k} = \frac{L_g(Y(i), Y(k))}{\sum_{k' \in \Omega} L_g(Y(i), Y(k'))} \text{ (dependency on } Y\text{)}$$

- Use only photometric proximity
- L: kernel and g: window size / smoothing parameter

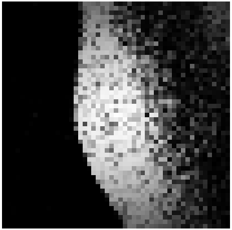
Some classical filters: $\widehat{I}(i) = \sum_k \theta_{i,k} Y(k)$



$\theta_{i,k}$: Gaussian



$\theta_{i,k}$: Yaroslavsky



$\theta_{i,k}$: Bilateral

Bilateral filter - Tomasi and Manduchi (98)

$$\theta_{i,k} = \frac{K_h(i,k)L_g(Y(i),Y(k))}{\sum_{k' \in \Omega} K_h(i,k')L_g(Y(i),Y(k'))}$$

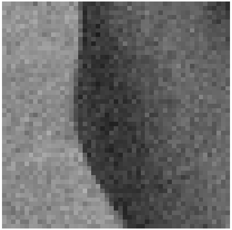
- Use spatial and photometric proximities.
- K, L: kernels and h, g: windows sizes / smoothing parameters

Data adaptive kernel

Examples:

- Yaroslavski and bilateral filters.
- $\star$-let thresholding (complex dependency of the weights...)
- ...

Data adaptive kernel

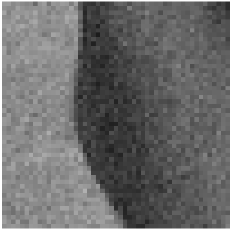


Original Image

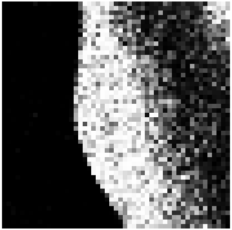
Examples:

- Yaroslavski and bilateral filters.
- $\star$-let thresholding (complex dependency of the weights...)
- ...

Data adaptive kernel



Original Image



$\theta_{i,k}$: Yaroslavsky

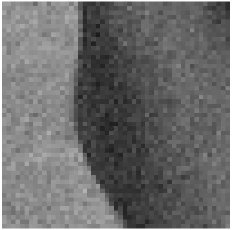
Examples:

- Yaroslavski and bilateral filters.
- $\star$-let thresholding (complex dependency of the weights...)
- ...

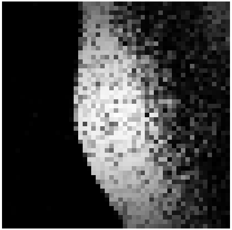
Intuition

- Intuition: average pixels close in both distance and value.
- Issue: pixel value = too local...

Data adaptive kernel



Original Image



$\theta_{i,k}$: Bilateral

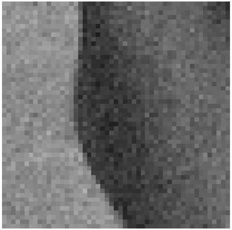
Examples:

- Yaroslavski and bilateral filters.
- $\star$-let thresholding (complex dependency of the weights...)
- ...

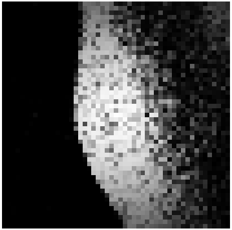
Intuition

- Intuition: average pixels close in both distance and value.
- Issue: pixel value = too local...

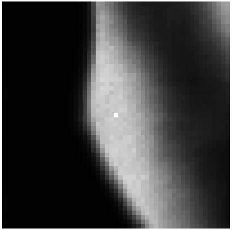
Data adaptive kernel



Original Image



$\theta_{i,k}$: Bilateral



$\theta_{i,k}$: NL-Means

Examples:

- Yaroslavski and bilateral filters.
- $\star$-let thresholding (complex dependency of the weights...)
- ...

Intuition

- Intuition: average pixels close in both distance and value.
- Issue: pixel value = too local...

Patches

Patch

- Patch: less local version of pixel value.
- Patch $P^Y(i)$: small image $P^Y(i)[\delta] = Y(i + \delta)$ with $\delta \in V_W$.
- Example: square $V_W = \{-W_- \leq \delta_1, \delta_2 \leq W_+\}$ with $W = W_- + W_+ + 1$.

Patches

Patch

- Patch: less local version of pixel value.
- Patch $P^Y(i)$: small image $P^Y(i)[\delta] = Y(i+\delta)$ with $\delta \in V_W$.
- Example: square $V_W = \{-W_- \leq \delta_1, \delta_2 \leq W_+\}$ with $W = W_- + W_+ + 1$.

Patch and images

- Operator $Y \mapsto P^Y$ sends the image $Y = (Y(i))$ to the patch collection $P^Y = (P^Y(i))$.
- Lifting of a 1-D image to a W^2-D image.
- *Easy* (naive) reprojection of a patch collection P^Y to an image Y...

Patches

Patch

- Patch: less local version of pixel value.
- Patch $P^Y(i)$: small image $P^Y(i)[\delta] = Y(i+\delta)$ with $\delta \in V_W$.
- Example: square $V_W = \{-W_- \leq \delta_1, \delta_2 \leq W_+\}$ with $W = W_- + W_+ + 1$.

Patch and images

- Operator $Y \mapsto P^Y$ sends the image $Y = (Y(i))$ to the patch collection $P^Y = (P^Y(i))$.
- Lifting of a 1-D image to a W^2-D image.
- *Easy* (naive) reprojection of a patch collection P^Y to an image Y...

Patch and estimation

- Estimation problem: estimation of the patch collection P^I from the observed patch collection $P^Y = P^I + \sigma P^{\mathcal{E}}$.
- Very different problem than the estimation of P^I from $P^I + \sigma \mathcal{E}$.

Patches

Patch

- Patch: less local version of pixel value.
- Patch $P^Y(i)$: small image $P^Y(i)[\delta] = Y(i+\delta)$ with $\delta \in V_W$.
- Example: square $V_W = \{-W_- \leq \delta_1, \delta_2 \leq W_+\}$ with $W = W_- + W_+ + 1$.

Patch and images

- Operator $Y \mapsto P^Y$ sends the image $Y = (Y(i))$ to the patch collection $P^Y = (P^Y(i))$.
- Lifting of a 1-D image to a W^2-D image.
- *Easy* (naive) reprojection of a patch collection P^Y to an image Y...

Patch and estimation

- Estimation problem: estimation of the patch collection P^I from the observed patch collection $P^Y = P^I + \sigma P^{\mathcal{E}}$.
- Very different problem than the estimation of P^I from $P^I + \sigma \mathcal{E}$.

Patch based methods

Kernel methods and patches

- Estimation by patch averaging:

$$\widehat{P^I}(i)[\delta] = \sum_k \theta_{i,k} P^Y(k)[\delta].$$

- If the weights are independent of P^Y, the patch structure is useless...

Patch based methods

Kernel methods and patches

- Estimation by patch averaging:

$$\widehat{P^I}(i)[\delta] = \sum_k \theta_{i,k} P^Y(k)[\delta].$$

- If the weights are independent of P^Y, the patch structure is useless...

Intuition

- Uses some weights which take into account the patch similarity:

Patches P_k^Y:

- Patch P_i^Y to denoise,
- Similar patches: useful $\rightarrow$ large weights,
- Less similar patches: less useful $\rightarrow$ smaller weights,
- Very different patches: useless $\rightarrow$ quasi null weights.

Patch based methods

Kernel methods and patches

- Estimation by patch averaging:

$$\widehat{P^I}(i)[\delta] = \sum_k \theta_{i,k} P^Y(k)[\delta].$$

- If the weights are independent of P^Y, the patch structure is useless...

Intuition

- Uses some weights which take into account the patch similarity:

Patches P_k^Y:

- Patch P_i^Y to denoise,
- Similar patches: useful $\rightarrow$ large weights,
- Less similar patches: less useful $\rightarrow$ smaller weights,
- Very different patches: useless $\rightarrow$ quasi null weights.

NL-Means

NL-Means (Buadès, Coll and Morel)

- Choose a dissimilarity measure between patches.
- Use weights $\theta_{i,k} = \dfrac{K'(D(P^Y(i), P^Y(k)))}{\sum_{k'} K'(D(P^Y(i), P^Y(k')))}$
- Choose $D(P^Y(i), P^Y(k)) = \|P^Y(i) - P^Y(k)\|_2$ as a dissimilarity measure, a Gaussian kernel $K'(x) = \exp(-x^2/\beta)$ and a temperature $\beta = \gamma\sigma^2$.

NL-Means

NL-Means (Buadès, Coll and Morel)

- Choose a dissimilarity measure between patches.
- Use weights $\theta_{i,k} = \frac{K'(D(P^Y(i), P^Y(k)))}{\sum_{k'} K'(D(P^Y(i), P^Y(k')))}$
- Choose $D(P^Y(i), P^Y(k)) = \|P^Y(i) - P^Y(k)\|_2$ as a dissimilarity measure, a Gaussian kernel $K'(x) = \exp(-x^2/\beta)$ and a temperature $\beta = \gamma\sigma^2$.

Results

- Fast and efficient method.
- State of the art method are variations around this principle.

NL-Means

NL-Means (Buadès, Coll and Morel)

- Choose a dissimilarity measure between patches.
- Use weights $\theta_{i,k} = \dfrac{K'(D(P^Y(i), P^Y(k)))}{\sum_{k'} K'(D(P^Y(i), P^Y(k')))}$
- Choose $D(P^Y(i), P^Y(k)) = \|P^Y(i) - P^Y(k)\|_2$ as a dissimilarity measure, a Gaussian kernel $K'(x) = \exp(-x^2/\beta)$ and a temperature $\beta = \gamma\sigma^2$.

Results

- Fast and efficient method.
- State of the art method are variations around this principle.

Variations

- Automatic adaptation of the search zone. (Kervrann et al.)
- Higher order local approximation. (Buadès et al.)
- Use of different similarity measure. (Guichard et al.)

NL-Means

NL-Means (Buadès, Coll and Morel)

- Choose a dissimilarity measure between patches.
- Use weights $\theta_{i,k} = \dfrac{K'(D(P^Y(i), P^Y(k)))}{\sum_{k'} K'(D(P^Y(i), P^Y(k')))}$
- Choose $D(P^Y(i), P^Y(k)) = \|P^Y(i) - P^Y(k)\|_2$ as a dissimilarity measure, a Gaussian kernel $K'(x) = \exp(-x^2/\beta)$ and a temperature $\beta = \gamma\sigma^2$.

Results

- Fast and efficient method.
- State of the art method are variations around this principle.

Variations

- Automatic adaptation of the search zone. (Kervrann et al.)
- Higher order local approximation. (Buadès et al.)
- Use of different similarity measure. (Guichard et al.)

BM3D

State-of-the-art

- Patch with adapted shapes, efficient patch denoising and clever patch reprojection.
- Dabov, Foi, Katkovnik and Egiazarian, *Image denoising by sparse 3-D transform-domain collaborative filtering* (2007) and *A non-local and shape-adaptive transform-domain collaborative filtering* (2008)
- Still best numerical results available...
- Matlab code at `http://www.cs.tut.fi/~foi/GCF-BM3D/`.

BM3D

State-of-the-art

- Patch with adapted shapes, efficient patch denoising and clever patch reprojection.
- Dabov, Foi, Katkovnik and Egiazarian, *Image denoising by sparse 3-D transform-domain collaborative filtering* (2007) and *A non-local and shape-adaptive transform-domain collaborative filtering* (2008)
- Still best numerical results available...
- Matlab code at `http://www.cs.tut.fi/~foi/GCF-BM3D/`.

Denoising with patches in three steps

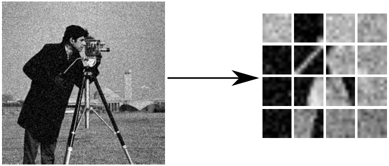
The 3 steps

Denoising with patches in three steps

The 3 steps

- Patchization: $Y = (Y(i)) \mapsto P^Y = (P^Y(i))$.

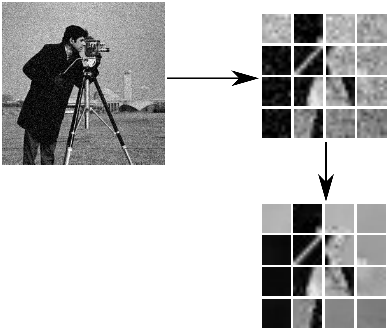
Denoising with patches in three steps



The 3 steps

- Patchization: $Y = (Y(i)) \mapsto P^Y = (P^Y(i))$.
- Patch estimation: $P^Y = (P^Y(i)) \mapsto \widehat{P^I} = (\widehat{P^I}(i))$.

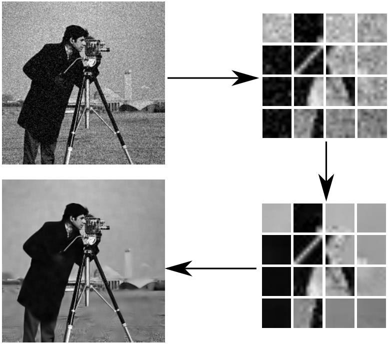
Denoising with patches in three steps



The 3 steps

- Patchization: $Y = (Y(i)) \mapsto P^Y = (P^Y(i))$.
- Patch estimation: $P^Y = (P^Y(i)) \mapsto \widehat{P^I} = (\widehat{P^I}(i))$.
- Patch reprojection: $\widehat{P^I} = (\widehat{P^I}(i)) \mapsto \widehat{I} = (\widehat{I}(i))$.

Denoising with patches in three steps



The 3 steps

- Patchization: $Y = (Y(i)) \mapsto P^Y = (P^Y(i))$.
- Patch estimation: $P^Y = (P^Y(i)) \mapsto \widehat{P^I} = (\widehat{P^I}(i))$.
- Patch reprojection: $\widehat{P^I} = (\widehat{P^I}(i)) \mapsto \widehat{I} = (\widehat{I}(i))$.

Some choices for the three steps

Patchization

- Square patch (NL-Means, BM3D, Mairal...)
- *Funny* (adapted) shape (BM3D, Salmon...)

Some choices for the three steps

Patchization

- Square patch (NL-Means, BM3D, Mairal...)
- *Funny* (adapted) shape (BM3D, Salmon...)

Patch estimation

- Patch averaging (NL-Means, Salmon),
- Patch grouping and adapted transform (wavelet, dct,...) (BM3D),
- Patch grouping and local dictionaries learning (Mairal, Salmon),
- Further aggregation... (BM3D, Salmon)

Some choices for the three steps

Patchization

- Square patch (NL-Means, BM3D, Mairal...)
- *Funny* (adapted) shape (BM3D, Salmon...)

Patch estimation

- Patch averaging (NL-Means, Salmon),
- Patch grouping and adapted transform (wavelet, dct,...) (BM3D),
- Patch grouping and local dictionaries learning (Mairal, Salmon),
- Further aggregation... (BM3D, Salmon)

Patch reprojection

- Naive reprojection or uniform averaging (NL-Means, Mairal).
- Clever aggregation of available pixel estimates (BM3D, Salmon)

Some choices for the three steps

Patchization

- Square patch (NL-Means, BM3D, Mairal...)
- *Funny* (adapted) shape (BM3D, Salmon...)

Patch estimation

- Patch averaging (NL-Means, Salmon),
- Patch grouping and adapted transform (wavelet, dct,...) (BM3D),
- Patch grouping and local dictionaries learning (Mairal, Salmon),
- Further aggregation... (BM3D, Salmon)

Patch reprojection

- Naive reprojection or uniform averaging (NL-Means, Mairal).
- Clever aggregation of available pixel estimates (BM3D, Salmon)

Secret of BM3D

- Very good choice (methods and parameters) for each step!

Some choices for the three steps

Patchization

- Square patch (NL-Means, BM3D, Mairal...)
- *Funny* (adapted) shape (BM3D, Salmon...)

Patch estimation

- Patch averaging (NL-Means, Salmon),
- Patch grouping and adapted transform (wavelet, dct,...) (BM3D),
- Patch grouping and local dictionaries learning (Mairal, Salmon),
- Further aggregation... (BM3D, Salmon)

Patch reprojection

- Naive reprojection or uniform averaging (NL-Means, Mairal).
- Clever aggregation of available pixel estimates (BM3D, Salmon)

Secret of BM3D

- Very good choice (methods and parameters) for each step!

Outline

1. A brief survey of patch based method
2. A survey of patch based estimation
3. Statistical aggregation
   - Initial estimates and aggregation
   - PAC-Bayesian aggregation
4. Patches and Aggregation

Initial estimates and aggregation

Model and initial estimates

- $Y = I + \sigma\mathcal{E}$ of size $W \times W$.
- Collection $\{P_k\}$ of M initial estimates of I.
- Linear combination $P_\theta = \sum_k \theta_k P_k$.

Initial estimates and aggregation

Model and initial estimates

- $Y = I + \sigma\mathcal{E}$ of size $W \times W$.
- Collection $\{P_k\}$ of M initial estimates of I.
- Linear combination $P_\theta = \sum_k \theta_k P_k$.

Aggregation

- Estimate I as a linear combination: $\widehat{I} = P_{\widehat{\theta}}$.
- Aggregation procedure $\simeq$ way of choosing $\widehat{\theta}$ from Y.

Initial estimates and aggregation

Model and initial estimates

- $Y = I + \sigma\mathcal{E}$ of size $W \times W$.
- Collection $\{P_k\}$ of M initial estimates of I.
- Linear combination $P_\theta = \sum_k \theta_k P_k$.

Aggregation

- Estimate I as a linear combination: $\widehat{I} = P_{\widehat{\theta}}$.
- Aggregation procedure $\simeq$ way of choosing $\widehat{\theta}$ from Y.

Oracle type inequality

- Quadratic risk: $R_\theta = \mathbb{E}\left(\|I - P_\theta\|^2\right)$.
- Typical result: *optimal* aggregation on a class Θ,

$$\mathbb{E}\left(\|I - \widehat{I}\|^2\right) \leq C \inf_{\theta \in \Theta} R_\theta + \mathrm{price}(\sigma^2, \theta)$$

- C, θ and price depend on the procedure.

Initial estimates and aggregation

Model and initial estimates

- $Y = I + \sigma\mathcal{E}$ of size $W \times W$.
- Collection $\{P_k\}$ of M initial estimates of I.
- Linear combination $P_\theta = \sum_k \theta_k P_k$.

Aggregation

- Estimate I as a linear combination: $\widehat{I} = P_{\widehat{\theta}}$.
- Aggregation procedure $\simeq$ way of choosing $\widehat{\theta}$ from Y.

Oracle type inequality

- Quadratic risk: $R_\theta = \mathbb{E}\left(\|I - P_\theta\|^2\right)$.
- Typical result: *optimal* aggregation on a class Θ,

$$\mathbb{E}\left(\|I - \widehat{I}\|^2\right) \leq C \inf_{\theta\in\Theta} R_\theta + \mathrm{price}(\sigma^2, \theta)$$

- C, θ and price depend on the procedure.

Aggregation procedures 1

Fixed strategy

- Choose the strategy without even looking at the observation...
- The set Θ is a singleton $\Leftrightarrow$ trivial oracle inequality!
- Classical choices:
  - Fixed choice $\Theta = e_{k_0}$.
  - Uniform average $\Theta = (1/M, \dots, 1/M)$.
  - Fixed weighted average...

Aggregation procedures 1

Fixed strategy

- Choose the strategy without even looking at the observation...
- The set Θ is a singleton $\Leftrightarrow$ trivial oracle inequality!
- Classical choices:
  - Fixed choice $\Theta = e_{k_0}$.
  - Uniform average $\Theta = (1/M, \ldots, 1/M)$.
  - Fixed weighted average...

Variance driven strategy

- Requirement: availability of $\hat{V}_k$ an estimate of the variance of P_k.
- Classical choices:
  - Choose $\widehat{\theta} = e_k$ with $k = \arg\min \hat{V}_k$.
  - Use an approximate error independent assumption and set $\widehat{\theta} \propto \left(\hat{V}_1^{-1}, \ldots, \hat{V}_1^{-1}\right)$ (Stacked Generalization - Wolpert (92)).
- If $\hat{V}_\theta$ an estimate of the variance of P_θ is available: $\widehat{\theta} = \arg\min \hat{V}_\theta$.
- Implicit negligible bias assumption.

Aggregation procedures 1

Fixed strategy

- Choose the strategy without even looking at the observation...
- The set Θ is a singleton $\Leftrightarrow$ trivial oracle inequality!
- Classical choices:
  - Fixed choice $\Theta = e_{k_0}$.
  - Uniform average $\Theta = (1/M, \ldots, 1/M)$.
  - Fixed weighted average...

Variance driven strategy

- Requirement: availability of $\hat{V}_k$ an estimate of the variance of P_k.
- Classical choices:
  - Choose $\widehat{\theta} = e_k$ with $k = \arg\min \hat{V}_k$.
  - Use an approximate error independent assumption and set $\widehat{\theta} \propto \left(\hat{V}_1^{-1}, \ldots, \hat{V}_1^{-1}\right)$ (Stacked Generalization - Wolpert (92)).
- If $\hat{V}_\theta$ an estimate of the variance of P_θ is available: $\widehat{\theta} = \arg\min \hat{V}_\theta$.
- Implicit negligible bias assumption.

Aggregation procedures 2

Unbiased Risk Estimate

- Requirement: availability of $\widehat{r}_\theta$ such that $\mathbb{E}\widehat{r}_\theta = \mathbb{E}\left(\|I - P_\theta\|^2\right) = R_\theta$.
- Selection by empirical risk minimization:

$$\widehat{I} = \arg\min_{P_\theta} \widehat{r}_\theta$$

- Oracle inequality under strong assumptions...

Aggregation procedures 2

Unbiased Risk Estimate

- Requirement: availability of $\widehat{r}_\theta$ such that $\mathbb{E}\widehat{r}_\theta = \mathbb{E}\left(\|I - P_\theta\|^2\right) = R_\theta$.
- Selection by empirical risk minimization:

$$\widehat{I} = \arg\min_{P_\theta} \widehat{r}_\theta$$

- Oracle inequality under strong assumptions...

Penalizations

- AIC/BIC: Selection by penalization proportional to the dimension:

$$\widehat{I} = \arg\min_{P_\theta} \widehat{r}_\theta + \lambda\|\theta\|_0.$$

  - Oracle inequality when P_k are fixed or obtained by projection.
  - Complex (NP-hard) numerical optimization.
- Lasso: Selection by penalization proportional to the ℓ^1 norm:

$$\widehat{I} = \arg\min_{P_\theta} \widehat{r}_\theta + \lambda\|\theta\|_1.$$

  - Oracle inequality for fixed P_k:
    - ℓ^0 price with strong assumptions,
    - ℓ^1 price without assumptions.
- Lepski's method...

Aggregation procedures 2

Unbiased Risk Estimate

- Requirement: availability of $\widehat{r}_\theta$ such that $\mathbb{E}\widehat{r}_\theta = \mathbb{E}\left(\|I - P_\theta\|^2\right) = R_\theta$.
- Selection by empirical risk minimization:

$$\widehat{I} = \arg\min_{P_\theta} \widehat{r}_\theta$$

- Oracle inequality under strong assumptions...

Penalizations

- AIC/BIC: Selection by penalization proportional to the dimension:

$$\widehat{I} = \arg\min_{P_\theta} \widehat{r}_\theta + \lambda\|\theta\|_0.$$

  - Oracle inequality when P_k are fixed or obtained by projection.
  - Complex (NP-hard) numerical optimization.
- Lasso: Selection by penalization proportional to the ℓ^1 norm:

$$\widehat{I} = \arg\min_{P_\theta} \widehat{r}_\theta + \lambda\|\theta\|_1.$$

  - Oracle inequality for fixed P_k:
    - ℓ^0 price with strong assumptions,
    - ℓ^1 price without assumptions.
- Lepski's method...

Random measure aggregation

Random measure ρ

- Credo: Mixing is better than selecting...
- Define the estimate $\widehat{l}$ through a mixing measure ρ on $\Theta \subset \mathbb{R}^M$:

$$\widehat{l} = \int_{\theta \in \mathbb{R}^M} P_\theta d\rho(\theta). \quad \text{(Randomized version possible)}$$

- As $\theta \mapsto f_\theta$ is linear, $\widehat{\theta} = \int_{\theta \in \Theta} \theta d\rho(\theta)$.
- If Θ is not convex, $\widehat{\theta}$ may not belongs to Θ!

Random measure aggregation

Random measure ρ

- Credo: Mixing is better than selecting...
- Define the estimate $\widehat{l}$ through a mixing measure ρ on $\Theta \subset \mathbb{R}^M$:

$$\widehat{l} = \int_{\theta \in \mathbb{R}^M} P_\theta d\rho(\theta). \quad \text{(Randomized version possible)}$$

- As $\theta \mapsto f_\theta$ is linear, $\widehat{\theta} = \int_{\theta \in \Theta} \theta d\rho(\theta)$.
- If Θ is not convex, $\widehat{\theta}$ may not belongs to Θ!

Example: Exponential weights

- Measure ρ supported on the e_k with exponential weighting scheme:

$$\rho \propto \sum_k e^{-\frac{1}{\beta}\widehat{r}_k} \delta_{e_k}.$$

- Resulting estimate: $\widehat{l} = \frac{1}{Z} \sum_k e^{-\frac{1}{\beta}\widehat{r}_k} P_k \ldots$

Random measure aggregation

Random measure ρ

- Credo: Mixing is better than selecting...
- Define the estimate $\widehat{I}$ through a mixing measure ρ on $\Theta \subset \mathbb{R}^M$:

$$\widehat{I} = \int_{\theta \in \mathbb{R}^M} P_\theta d\rho(\theta). \quad \text{(Randomized version possible)}$$

- As $\theta \mapsto f_\theta$ is linear, $\widehat{\theta} = \int_{\theta \in \Theta} \theta d\rho(\theta)$.
- If Θ is not convex, $\widehat{\theta}$ may not belongs to Θ!

Example: Exponential weights

- Measure ρ supported on the e_k with exponential weighting scheme:

$$\rho \propto \sum_k e^{-\frac{1}{\beta}\widehat{r_k}} \delta_{e_k}.$$

- Resulting estimate: $\widehat{I} = \frac{1}{Z} \sum_k e^{-\frac{1}{\beta}\widehat{r_k}} P_k \ldots$

Generalized exponential weights

Prior

- Prior law π on $\Theta \subset \mathbb{R}^M$.
- Different role than a Bayesian prior.

Generalized exponential weights

Prior

- Prior law π on $\Theta \subset \mathbb{R}^M$.
- Different role than a Bayesian prior.

Generalized exponential weight aggregation

- Specific PAC-Bayesian procedure.
- ρ depends on π and on a temperature β:

$$d\rho(\theta) = \frac{e^{-\frac{1}{\beta}\widehat{r}_\theta}}{\int_{\mathbb{R}^M} e^{-\frac{1}{\beta}\widehat{r}_{\theta'}}\, d\pi(\theta')} d\pi(\theta) \quad .$$

- Estimate: $\widehat{l} = \int_{\mathbb{R}^M} \frac{e^{-\frac{1}{\beta}\widehat{r}_\theta}}{\int_{\mathbb{R}^M} e^{-\frac{1}{\beta}\widehat{r}_{\theta'}}\, d\pi(\theta')} P_\theta d\pi(\theta)$
- If $\pi = \sum_k \pi_k \delta_{e_k}$: $\widehat{l} = \sum_k \frac{e^{-\frac{1}{\beta}\widehat{r}_\theta} \pi_k}{\sum_{k'} e^{-\frac{1}{\beta}\widehat{r}_{k'}} \pi_{k'}} P_k \quad .$

Generalized exponential weights

Prior

- Prior law π on $\Theta \subset \mathbb{R}^M$.
- Different role than a Bayesian prior.

Generalized exponential weight aggregation

- Specific PAC-Bayesian procedure.
- ρ depends on π and on a temperature β:
$$d\rho(\theta) = \frac{e^{-\frac{1}{\beta}\widehat{r_\theta}}}{\int_{\mathbb{R}^M} e^{-\frac{1}{\beta}\widehat{r_{\theta'}}} d\pi(\theta')} d\pi(\theta) \quad .$$
- Estimate: $\widehat{I} = \int_{\mathbb{R}^M} \frac{e^{-\frac{1}{\beta}\widehat{r_\theta}}}{\int_{\mathbb{R}^M} e^{-\frac{1}{\beta}\widehat{r_{\theta'}}} d\pi(\theta')} P_\theta d\pi(\theta)$
- If $\pi = \sum_k \pi_k \delta_{e_k}$: $\widehat{I} = \sum_k \frac{e^{-\frac{1}{\beta}\widehat{r_\theta}} \pi_k}{\sum_{k'} e^{-\frac{1}{\beta}\widehat{r_{k'}}} \pi_{k'}} P_k \quad .$

Theoretical results

Assumptions on P_k

- No general results...
- P_k obtained by projection and Gaussian noise (Barron, Leung).
- P_k independents of Y and various noise assumption (Dalalyan, Tsybakov).

Theoretical results

Assumptions on P_k

- No general results...
- P_k obtained by projection and Gaussian noise (Barron, Leung).
- P_k independents of Y and various noise assumption (Dalalyan, Tsybakov).

Oracle inequality

- *Sharp* oracle inequality: if $\beta \geq 4\sigma^2$,

$$\mathbb{E}\left(\|I - \widehat{I}\|^2\right) \leq \inf_p \int_{\theta \in \mathbb{R}^M} \mathbb{E}\left(\|I - P_\theta\|^2\right) dp(\theta) + \beta \mathcal{K}(p, \pi)$$

where $\mathcal{K}(p, \pi)$ is the Kullback-Leibler divergence

$$\mathcal{K}(p, \pi) = \begin{cases} \mathbb{E}_p\left(\log\left(\frac{dp}{d\pi}\right)\right) & \text{if } p \ll \pi \\ +\infty & \text{otherwise} \end{cases}.$$

Theoretical results

Assumptions on P_k

- No general results...
- P_k obtained by projection and Gaussian noise (Barron, Leung).
- P_k independents of Y and various noise assumption (Dalalyan, Tsybakov).

Oracle inequality

- *Sharp* oracle inequality: if $\beta \geq 4\sigma^2$,

$$\mathbb{E}\left(\|I - \widehat{I}\|^2\right) \leq \inf_p \int_{\theta \in \mathbb{R}^M} \mathbb{E}\left(\|I - P_\theta\|^2\right) dp(\theta) + \beta \mathcal{K}(p, \pi)$$

where $\mathcal{K}(p, \pi)$ is the Kullback-Leibler divergence

$$\mathcal{K}(p, \pi) = \begin{cases} \mathbb{E}_p\left(\log\left(\frac{dp}{d\pi}\right)\right) & \text{if } p \ll \pi \\ +\infty & \text{otherwise} \end{cases}.$$

Prior choice

Error bound and prior

- $\mathbb{E}\left(\|I-\widehat{I}\|^2\right) \leq \inf_p \int_{\theta\in\mathbb{R}^M} \mathbb{E}\left(\|I-P_\theta\|^2\right) dp(\theta) + \beta\mathcal{K}(p,\pi)$
- Trade-off between a p concentrated around the θ of best P_θ and a p close to π.
- Choice of π so that this quantity is *always* small...

Prior choice

Error bound and prior

- $\mathbb{E}\left(\|I - \widehat{I}\|^2\right) \leq \inf_p \int_{\theta \in \mathbb{R}^M} \mathbb{E}\left(\|I - P_\theta\|^2\right) dp(\theta) + \beta \mathcal{K}(p, \pi)$
- Trade-off between a p concentrated around the θ of best P_θ and a p close to π.
- Choice of π so that this quantity is *always* small...

Discrete prior

- $\pi = \sum_k \pi_k \delta_{e_k}$: $\mathbb{E}\left(\|I - \widehat{I}\|^2\right) \leq \inf_k \mathbb{E}\left(\|I - P_k\|^2\right) + \beta |\log \pi_k|$.
- Better than the best initial estimate up to a penalty: model selection.

Prior choice

Error bound and prior

- $\mathbb{E}\left(\|I-\widehat{I}\|^2\right) \leq \inf_p \int_{\theta\in\mathbb{R}^M} \mathbb{E}\left(\|I-P_\theta\|^2\right) dp(\theta) + \beta\mathcal{K}(p,\pi)$
- Trade-off between a p concentrated around the θ of best P_θ and a p close to π.
- Choice of π so that this quantity is *always* small...

Discrete prior

- $\pi = \sum_k \pi_k \delta_{e_k}$: $\mathbb{E}\left(\|I-\widehat{I}\|^2\right) \leq \inf_k \mathbb{E}\left(\|I-P_k\|^2\right) + \beta|\log \pi_k|$.
- Better than the best initial estimate up to a penalty: model selection.

Sparse prior

- π: i.i.d. Student or Gaussian mixture (Dalalyan et al.).
- Bound: $\mathbb{E}(\|I-\widehat{I}\|^2) \leq \inf_\theta \mathbb{E}(\|I-P_\theta\|^2) + C\beta\|\theta\|_0 \log M$.
- Better than the best *sparse* aggregation...

Prior choice

Error bound and prior

- $\mathbb{E}\left(\|I-\widehat{I}\|^2\right) \leq \inf_p \int_{\theta\in\mathbb{R}^M} \mathbb{E}\left(\|I-P_\theta\|^2\right) dp(\theta) + \beta\mathcal{K}(p,\pi)$
- Trade-off between a p concentrated around the θ of best P_θ and a p close to π.
- Choice of π so that this quantity is *always* small...

Discrete prior

- $\pi = \sum_k \pi_k \delta_{e_k}$: $\mathbb{E}\left(\|I-\widehat{I}\|^2\right) \leq \inf_k \mathbb{E}\left(\|I-P_k\|^2\right) + \beta|\log \pi_k|$.
- Better than the best initial estimate up to a penalty: model selection.

Sparse prior

- π: i.i.d. Student or Gaussian mixture (Dalalyan et al.).
- Bound: $\mathbb{E}(\|I-\widehat{I}\|^2) \leq \inf_\theta \mathbb{E}(\|I-P_\theta\|^2) + C\beta\|\theta\|_0 \log M$.
- Better than the best *sparse* aggregation...

General PAC-Bayesian result

Loss and arbitrary random measure

- Previous results limited to fixed design loss and exponential weights.
- Generalization possible to random design and data depend estimate!

General PAC-Bayesian result

Loss and arbitrary random measure

- Previous results limited to fixed design loss and exponential weights.
- Generalization possible to random design and data depend estimate!

A general result (Catoni)

- Let $\theta_\star = \arg\min R_\theta$, if
  $\forall \theta, \mathbb{E}\left[e^{\frac{1}{\beta}[(R_\theta - R_{\theta_\star}) - (\widetilde{r}_\theta - \widehat{r}_{\theta_\star})]}\right] \leq e^{\left(1 - \frac{1}{C(\beta)}\right)(R_\theta - R_{\theta_\star})}$
  then for any prior π, with proba. $\geq 1 - \epsilon$, for any ρ
  $$\int_{\mathbb{R}^M} R_\theta d\rho(\theta) - R_{\theta_\star}$$
  $$\leq C(\beta)\left(\int_{\mathbb{R}^M} \widetilde{r}_\theta d\rho(\theta) - \widehat{r}_{\theta_\star} + \beta\left(\mathcal{K}(\rho, \pi) + \log(\epsilon^{-1})\right)\right) \quad .$$
- $\widetilde{r}_\theta$ can be a slight over estimation of $\widehat{r}_\theta$.
- Similar result in expectation.

General PAC-Bayesian result

Loss and arbitrary random measure

- Previous results limited to fixed design loss and exponential weights.
- Generalization possible to random design and data depend estimate!

A general result (Catoni)

- Let $\theta_\star = \arg\min R_\theta$, if
  $\forall\theta, \mathbb{E}\left[e^{\frac{1}{\beta}[(R_\theta - R_{\theta_\star}) - (\widetilde{r}_\theta - \widehat{r}_{\theta_\star})]}\right] \leq e^{\left(1-\frac{1}{C(\beta)}\right)(R_\theta - R_{\theta_\star})}$
  then for any prior π, with proba. $\geq 1-\epsilon$, for any ρ
  $$\int_{\mathbb{R}^M} R_\theta d\rho(\theta) - R_{\theta_\star} \leq C(\beta)\left(\int_{\mathbb{R}^M} \widetilde{r}_\theta d\rho(\theta) - \widehat{r}_{\theta_\star} + \beta\left(\mathcal{K}(\rho,\pi) + \log(\epsilon^{-1})\right)\right) \quad .$$
- $\widetilde{r}_\theta$ can be a slight over estimation of $\widehat{r}_\theta$.
- Similar result in expectation.

How to use it?

- Optimize the right side in ρ... $\rightarrow$ exponential weights!

General PAC-Bayesian result

Loss and arbitrary random measure

- Previous results limited to fixed design loss and exponential weights.
- Generalization possible to random design and data depend estimate!

A general result (Catoni)

- Let $\theta_\star = \arg\min R_\theta$, if
  $\forall \theta, \mathbb{E}\left[e^{\frac{1}{\beta}[(R_\theta - R_{\theta_\star}) - (\widetilde{r}_\theta - \widehat{r}_{\theta_\star})]}\right] \leq e^{\left(1-\frac{1}{C(\beta)}\right)(R_\theta - R_{\theta_\star})}$
  then for any prior π, with proba. $\geq 1-\epsilon$, for any ρ
  $$\int_{\mathbb{R}^M} R_\theta d\rho(\theta) - R_{\theta_\star} \leq C(\beta)\left(\int_{\mathbb{R}^M} \widetilde{r}_\theta d\rho(\theta) - \widehat{r}_{\theta_\star} + \beta\left(\mathcal{K}(\rho,\pi) + \log(\epsilon^{-1})\right)\right) \quad .$$
- $\widetilde{r}_\theta$ can be a slight over estimation of $\widehat{r}_\theta$.
- Similar result in expectation.

How to use it?

- Optimize the right side in ρ... $\rightarrow$ exponential weights!

Outline

1. A brief survey of patch based method
2. A survey of patch based estimation
3. Statistical aggregation
4. Patches and Aggregation
   - Framework and theory?
   - Patchwise aggregation
   - Pixelwise aggregation

Patch aggregation

Patches, aggregated estimate, SURE and prior

- Use patch $P^Y(i)$ as observation and M patches $P^Y(k)$ as initial estimates.
- Aggregated estimates: $P_\theta(i) = \sum_k \theta_{i,k} P^Y(k)$
- Unbiased estimate of the risk $\widehat{r}_\theta(i)$ (SURE) $\left(\mathbb{E}\left(\widehat{r}_\theta(i)\right) = \mathbb{E}\left(\|P^I(i) - P_\theta(i)\|^2\right)\right)$:

$$\widehat{r}_\theta(i) = \|P^Y(i) - P_\theta\|^2 - W^2(1 - 2\theta_{i,i})\sigma^2.$$

- Choice of a prior π and a temperature β.

Patch aggregation

Patches, aggregated estimate, SURE and prior

- Use patch $P^Y(i)$ as observation and M patches $P^Y(k)$ as initial estimates.
- Aggregated estimates: $P_\theta(i) = \sum_k \theta_{i,k} P^Y(k)$
- Unbiased estimate of the risk $\widehat{r}_\theta(i)$ (SURE) $\left(\mathbb{E}\left(\widehat{r}_\theta(i)\right) = \mathbb{E}\left(\|P^I(i) - P_\theta(i)\|^2\right)\right)$:

$$\widehat{r}_\theta(i) = \|P^Y(i) - P_\theta\|^2 - W^2(1 - 2\theta_{i,i})\sigma^2.$$

- Choice of a prior π and a temperature β.

Patch based exponential weight aggregation

- $\widehat{P^I}(i) = \int_{\mathbb{R}^M} \frac{e^{-\frac{1}{\beta}\widehat{r}_\theta(i)}}{\int_{\mathbb{R}^M} e^{-\frac{1}{\beta}\widehat{r}_{\theta'}(i)} d\pi(\theta')} P_\theta(i) d\pi(\theta)$.

Patch aggregation

Patches, aggregated estimate, SURE and prior

- Use patch $P^Y(i)$ as observation and M patches $P^Y(k)$ as initial estimates.
- Aggregated estimates: $P_\theta(i) = \sum_k \theta_{i,k} P^Y(k)$
- Unbiased estimate of the risk $\widehat{r}_\theta(i)$ (SURE) $\left(\mathbb{E}\left(\widehat{r}_\theta(i)\right) = \mathbb{E}\left(\|P^I(i) - P_\theta(i)\|^2\right)\right)$:

$$\widehat{r}_\theta(i) = \|P^Y(i) - P_\theta\|^2 - W^2(1 - 2\theta_{i,i})\sigma^2.$$

- Choice of a prior π and a temperature β.

Patch based exponential weight aggregation

- $\widehat{P^I}(i) = \int_{\mathbb{R}^M} \frac{e^{-\frac{1}{\beta}\widehat{r_\theta}(i)}}{\int_{\mathbb{R}^M} e^{-\frac{1}{\beta}\widehat{r}_{\theta'}(i)} d\pi(\theta')} P_\theta(i) d\pi(\theta)$.

Theoretical results

Theorem

- For $\beta \geq 4\sigma^2$,

$$\mathbb{E}\left(\|P^I(i) - \widehat{P^I}(i)\|^2\right) \leq C(\beta)\left(\inf_p \int_\theta \left(\|P^I(i) - P_\theta(i)\|^2 + W^2\sigma^2\|\theta\|^2\right) dp + \beta\mathcal{K}(p,\pi)\right)$$

Theoretical results

Theorem

- For $\beta \geq 4\sigma^2$,

$$\mathbb{E}\left(\|P^I(i) - \widehat{P^I}(i)\|^2\right) \leq C(\beta)\left(\inf_p \int_\theta \left(\|P^I(i) - P_\theta(i)\|^2 + W^2\sigma^2\|\theta\|^2\right) dp + \beta\mathcal{K}(p,\pi)\right)$$

Still work in progress

- $P_\theta(i)$ are neither projection based nor frozen $\Rightarrow$ big difficulties...
- Result valid with $C(\beta) = 1$ for a two independent observations model or with a pixel *splitting* strategy.
- Very specific form of the $P_\theta(i)$ and concentration of χ^2 $\Rightarrow$ exponential bound required by Catoni should hold.
- Loss in the constant (Catoni et al.) (and overestimation of $\widehat{r}_\theta$ (Barron and Leung)?).

Theoretical results

Theorem???

- For $\beta \geq 4\sigma^2$,

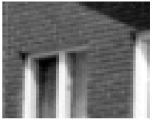
$$\mathbb{E}\left(\|P^I(i) - \widehat{P^I}(i)\|^2\right) \\ \leq C(\beta)\left(\inf_p \int_\theta \left(\|P^I(i) - P_\theta(i)\|^2 + W^2\sigma^2\|\theta\|^2\right) dp + \beta\mathcal{K}(p,\pi)\right)$$

Still work in progress

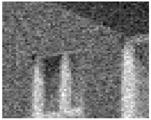
- $P_\theta(i)$ are neither projection based nor frozen $\Rightarrow$ big difficulties...
- Result valid with $C(\beta) = 1$ for a two independent observations model or with a pixel *splitting* strategy.
- Very specific form of the $P_\theta(i)$ and concentration of $\chi^2 \Rightarrow$ exponential bound required by Catoni should hold.
- Loss in the constant (Catoni et al.) (and overestimation of $\widehat{r}_\theta$ (Barron and Leung)?).

Some numerical results

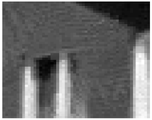
Some numerical results



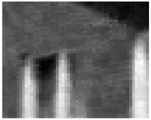
Original



Noisy (22.06 dB)



NL-Means (29.69 dB)

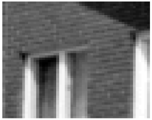


PAC-Bayesian (29.69 dB)

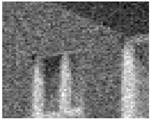
Methodology

- Comparison with NL-Means with a good temperature β.
- Patches PAC-Bayesian aggregation with Student prior.

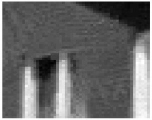
Some numerical results



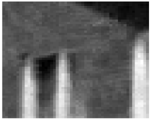
Original



Noisy (22.06 dB)



NL-Means (29.69 dB)



PAC-Bayesian (29.69 dB)

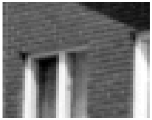
Methodology

- Comparison with NL-Means with a good temperature β.
- Patches PAC-Bayesian aggregation with Student prior.

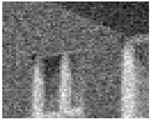
Results

- Similar to those obtained with NL-Means...
- $+$ parameter stability and room for improvement...

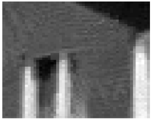
Some numerical results



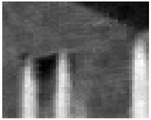
Original



Noisy (22.06 dB)



NL-Means (29.69 dB)



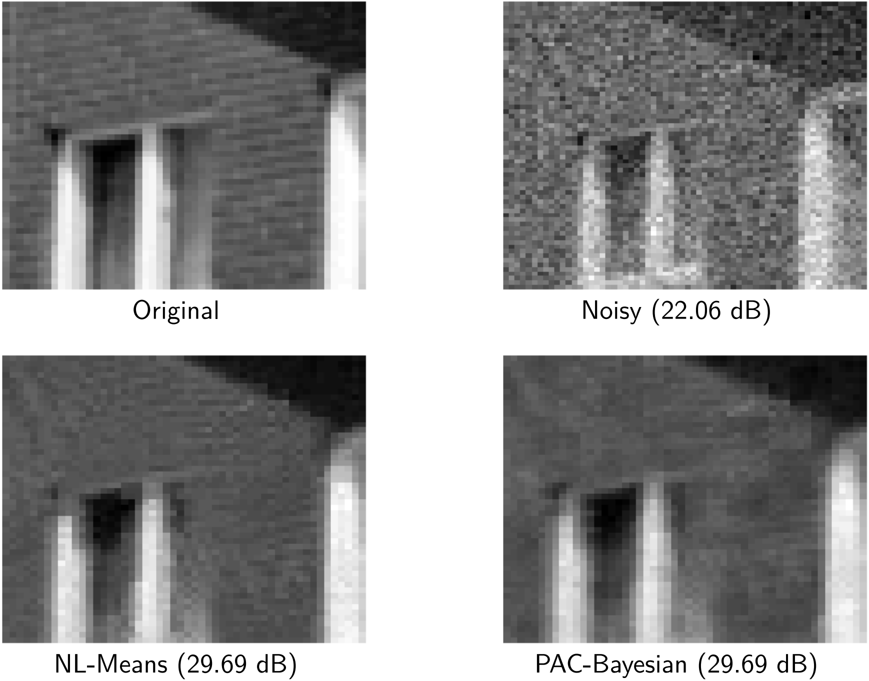
PAC-Bayesian (29.69 dB)

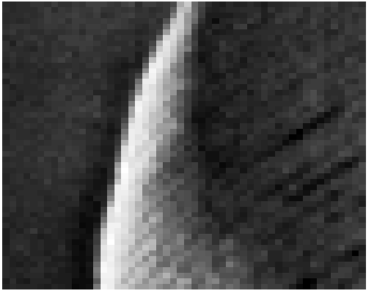
Methodology

- Comparison with NL-Means with a good temperature β.
- Patches PAC-Bayesian aggregation with Student prior.

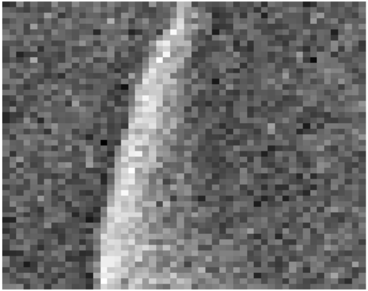
Results

- Similar to those obtained with NL-Means...
- $+$ parameter stability and room for improvement...

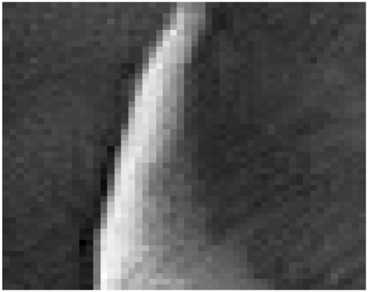

Original
Noisy (22.06 dB)
NL-Means (29.69 dB)
PAC-Bayesian (29.69 dB)



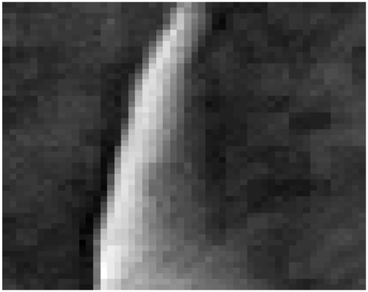
Original



Noisy (22.28 dB)

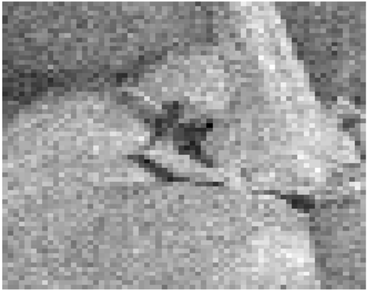


NL-Means (31.59 dB)



PAC-Bayesian (30.78 dB)

Original



Noisy (22.21 dB)

NL-Means (24.23dB)

PAC-Bayesian (26.96 dB)

Patchwise aggregation

Classical NL-Means and other patch estimates

- Patch estimation by aggregation of observed patches!
- Use of unbiased estimate of the risk in the weights (important for central patch weight).
- Other patch estimates are possible:
  - Oriented filtering,
  - Representation based approach (DCT, PCA, dictionary...)
  - IBR!!!
- For each patch $P^I(i)$, we can obtain a family of estimates $(\widehat{P^I}_k(i))$ (different parameter choices, different methods,...).

Patchwise aggregation

Classical NL-Means and other patch estimates

- Patch estimation by aggregation of observed patches!
- Use of unbiased estimate of the risk in the weights (important for central patch weight).
- Other patch estimates are possible:
  - Oriented filtering,
  - Representation based approach (DCT, PCA, dictionary...)
  - IBR!!!
- For each patch $P^I(i)$, we can obtain a family of estimates $(\widehat{P^I}_k(i))$ (different parameter choices, different methods,...).

How to combine them at a patch scale?

- Patchwise aggregation:
  - Minimum Variance Principle,
  - Unbiased Risk Minimizer / Penalized Risk Minimizer
  - Linear Combination and Risk Minimizer
  - PAC-Bayesian Aggregation...

Patchwise aggregation

Classical NL-Means and other patch estimates

- Patch estimation by aggregation of observed patches!
- Use of unbiased estimate of the risk in the weights (important for central patch weight).
- Other patch estimates are possible:
  - Oriented filtering,
  - Representation based approach (DCT, PCA, dictionary...)
  - IBR!!!
- For each patch $P^I(i)$, we can obtain a family of estimates $(\widehat{P^I}_k(i))$ (different parameter choices, different methods,...).

How to combine them at a patch scale?

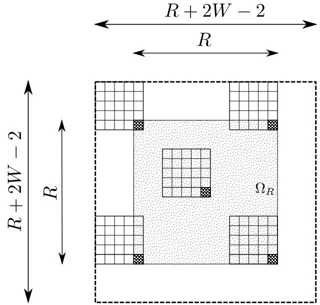
- Patchwise aggregation:
  - Minimum Variance Principle,
  - Unbiased Risk Minimizer / Penalized Risk Minimizer
  - Linear Combination and Risk Minimizer
  - PAC-Bayesian Aggregation...

Pixelwise aggregation

Patches and pixel

- A given pixel belongs to several patches!
- Several pixel values are available for each pixel...

Pixelwise aggregation



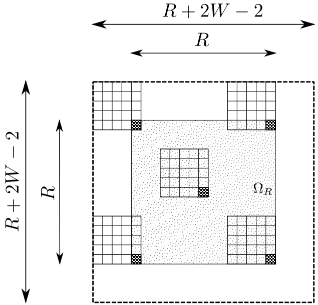
Patches and pixel

- A given pixel belongs to several patches!
- Several pixel values are available for each pixel...

Pixelwise aggregation

- How to combine those estimates?
- Naive solution: use only the central value of the patch centered on the pixel...

Pixelwise aggregation



Patches and pixel

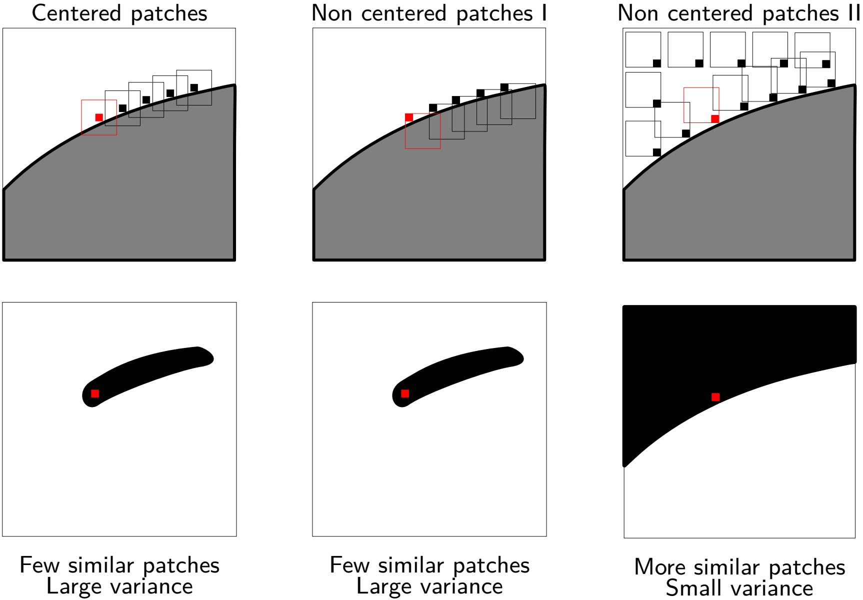
- A given pixel belongs to several patches!
- Several pixel values are available for each pixel...

Pixelwise aggregation

- How to combine those estimates?
- Naive solution: use only the central value of the patch centered on the pixel...

Some centering are better than the others...

Some centering are better than the others...



Variance based approach

Flat kernel

- Gaussian kernel can be replaced by a flat kernel.
- Similar numerical performance...
- For each centering, Var = Nb of similar patches^{-1}

Variance based approach

Flat kernel

- Gaussian kernel can be replaced by a flat kernel.
- Similar numerical performance...
- For each centering, Var = Nb of similar patches^{-1}

Pixelwise aggregation

- 2 variance based strategies compared:
  - Selection of the patch with the largest number of similar patches.
  - Average with weight $\propto$ Var^{-1} = Nb of similar patches
- Extension possible to Gaussian kernel.
- Similar patches $\simeq$ small bias $\Rightarrow$ Variance only approach makes sense.

Variance based approach

Flat kernel

- Gaussian kernel can be replaced by a flat kernel.
- Similar numerical performance...
- For each centering, Var $=$ Nb of similar patches^{-1}

Pixelwise aggregation

- 2 variance based strategies compared:
  - Selection of the patch with the largest number of similar patches.
  - Average with weight $\propto$ Var^{-1} $=$ Nb of similar patches
- Extension possible to Gaussian kernel.
- Similar patches $\simeq$ small bias $\Rightarrow$ Variance only approach makes sense.

Pixelwise aggregation

Pixelwise aggregation

(a)

(b)

(c)

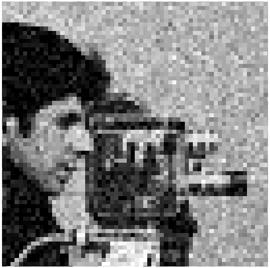
(d)

(e)

(a) Noisy
(b) Central
(c) Uniform
(d) Minimum variance
(e) Weight. Avg. $\propto \text{Var}^{-1}$
(Salmon et al.)

Pixelwise aggregation

(a)



(b)

(c)

(a) Noisy
(b) Central
(c) Uniform
(d) Minimum variance
(e) Weight. Avg. $\propto$ Var^{-1}
(Salmon et al.)

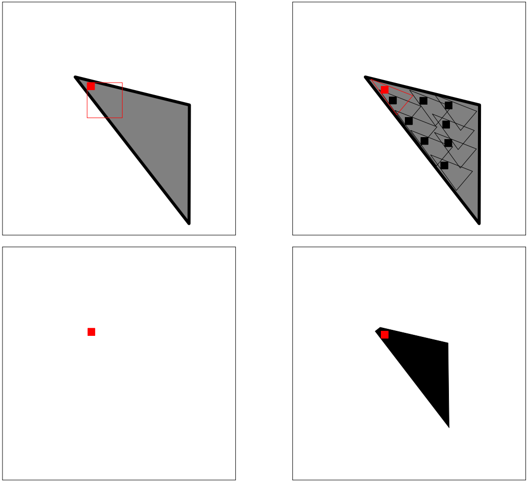
(d)

(e)

Some patch shapes are better than the others...

Some patch shapes are better than the others...

Square patch | Shape adapted patches



Few similar patches | More similar patches

But one need to choose..

But one need to choose..

SURE

Stein Unbiased Risk Estimate

- Gaussian noise: pixelwise unbiased risk $\widehat{r}_k(i)$ estimate available through Stein's formula.
- Extension possible for other noises...
- Explicit computation for NL-Means type estimates.

SURE

Stein Unbiased Risk Estimate

- Gaussian noise: pixelwise unbiased risk $\widehat{r}_k(i)$ estimate available through Stein's formula.
- Extension possible for other noises...
- Explicit computation for NL-Means type estimates.

Exponential Weight Aggregation

- Combine estimates with weights $\propto \exp\left(-\frac{1}{\beta}\widehat{r}_k(i)\right)$.
- Issue: pixelwise SURE estimates have large variances...
- Decision made on a regularized risk estimate (e.g. average on a small neighborhood)!

SURE

Stein Unbiased Risk Estimate

- Gaussian noise: pixelwise unbiased risk $\widehat{r}_k(i)$ estimate available through Stein's formula.
- Extension possible for other noises...
- Explicit computation for NL-Means type estimates.

Exponential Weight Aggregation

- Combine estimates with weights $\propto \exp\left(-\frac{1}{\beta}\widehat{r}_k(i)\right)$.
- Issue: pixelwise SURE estimates have large variances...
- Decision made on a regularized risk estimate (e.g. average on a small neighborhood)!

Salmon et al. ($\sigma = 20$)

Salmon et al. ($\sigma = 20$)

NL-Means

Weighted Average

EWA Mixed shapes

BM3D

Your Three-Step Denoising Program

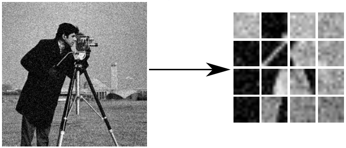
The 3 steps

Your Three-Step Denoising Program

The 3 steps

- Step 1 (Patchization): Construct a covering set of noisy patches with your favorite techniques.

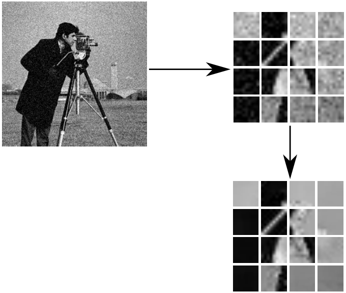
Your Three-Step Denoising Program



The 3 steps

- Step 1 (Patchization): Construct a covering set of noisy patches with your favorite techniques.
- Step 2 (Patch estimation): Produce estimates for each patches with your favorite techniques.

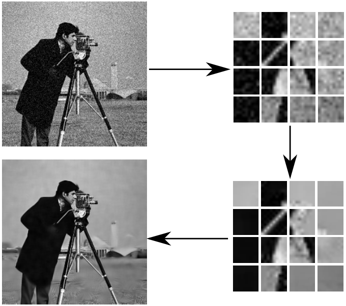
Your Three-Step Denoising Program



The 3 steps

- Step 1 (Patchization): Construct a covering set of noisy patches with your favorite techniques.
- Step 2 (Patch estimation): Produce estimates for each patches with your favorite techniques.
- Step 3 (Patch reprojection): Aggregate those estimates pixelwise with your favorite techniques.

Your Three-Step Denoising Program



The 3 steps

- Step 1 (Patchization): Construct a covering set of noisy patches with your favorite techniques.
- Step 2 (Patch estimation): Produce estimates for each patches with your favorite techniques.
- Step 3 (Patch reprojection): Aggregate those estimates pixelwise with your favorite techniques.

Conclusion

Brief survey of patch based method.

- Very subjective survey.
- Focused on denoising.
- Three-step denoising program.

Conclusion

Brief survey of patch based method.

- Very subjective survey.
- Focused on denoising.
- Three-step denoising program.

Three-Step Program

- Step 1: Split data in overlapping sets.
- Step 2: Use your favorite methods on these sets.
- Step 3: Aggregate the results on an individual basis.

Conclusion

Brief survey of patch based method.

- Very subjective survey.
- Focused on denoising.
- Three-step denoising program.

Three-Step Program

- Step 1: Split data in overlapping sets.
- Step 2: Use your favorite methods on these sets.
- Step 3: Aggregate the results on an individual basis.

Do you believe in this program?

- Patches? Processing? Aggregation?
- Efficiency? Complexity? Performance?

Conclusion

Brief survey of patch based method.

- Very subjective survey.
- Focused on denoising.
- Three-step denoising program.

Three-Step Program

- Step 1: Split data in overlapping sets.
- Step 2: Use your favorite methods on these sets.
- Step 3: Aggregate the results on an individual basis.

Do you believe in this program?

- Patches? Processing? Aggregation?
- Efficiency? Complexity? Performance?