Multiscale analysis in image processing

Inverse problems resolution

Barbara Pascal† and Nelly Pustelnik‡

bpascal-fr.github.io/talks

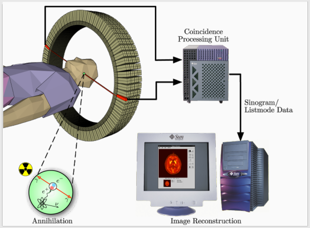
June 2025

† Nantes Université, École Centrale Nantes, CNRS, LS2N, F-44000 Nantes, France

‡ CNRS, ENSL, Laboratoire de physique, F-69342 Lyon, France

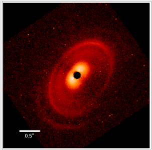
Computational imaging systems: examples

Medical imaging



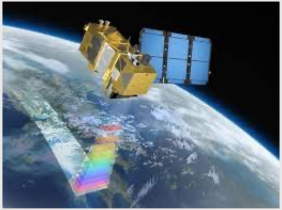
Astronomy

SPHERE/IRDIS

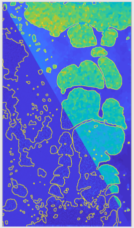


@ L. Denneulin

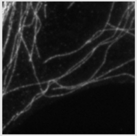
Remote sensing



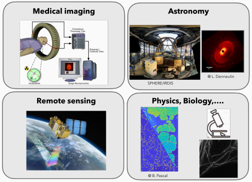
Physics, Biology,....



@ B. Pascal



Notations and basics



→ Variables of interest

- $z \in \mathbb{R}^M$: data/measurements.
- $\overline{x} \in \mathbb{R}^N$: unknown image.
- $\widehat{x} \in \mathbb{R}^N$: estimated image.

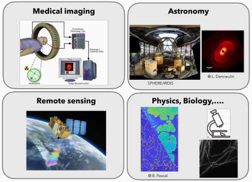
→ Forward model:

$$z = \mathcal{D}(A\overline{x})$$

Stochastic degradation ($\mathcal{D}$)

Linear operator (A)

Computational imaging systems: quantities of interest



➡ Variables of interest

- $\mathrm{z} \in \mathbb{R}^M$: data/measurements.
- $\overline{\mathrm{x}} \in \mathbb{R}^N$: unknown image.
- $\widehat{\mathrm{x}} \in \mathbb{R}^N$: estimated image.

➡ Forward model:

$$\mathrm{z} = \mathcal{D}(\mathrm{A}\overline{\mathrm{x}})$$

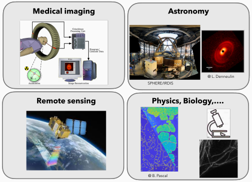
Stochastic degradation

Linear operator

➡ Inverse model:

$$\widehat{\mathrm{x}} = \mathrm{d}_{\Theta}(\mathrm{z})$$

Computational imaging systems: quantities of interest



➡ Variables of interest

- $\mathrm{z} \in \mathbb{R}^M$: data/measurements.
- $\overline{\mathrm{x}} \in \mathbb{R}^N$: unknown image.
- $\widehat{\mathrm{x}} \in \mathbb{R}^N$: estimated image.

➡ Forward model:

$$\mathrm{z} = \mathcal{D}(\mathrm{A}\overline{\mathrm{x}})$$

Stochastic degradation — **Linear** operator

➡ Inverse model:

$$\widehat{\mathrm{x}} = \mathrm{d}_\Theta(\mathrm{z})$$

➡ **Goal**: Estimate $\widehat{\mathrm{x}}$ close to $\overline{\mathrm{x}}$ from the information included in z, from the full or partial knowledge of A, from noise statistics $\mathcal{D}$, and from *a priori* knowledge on the class of image to recover.

Vector representation of an image

Image = matrix of pixels $\quad \mathrm{x} \in \mathbb{R}^{N_1 \times N_2}$

255	255	253	252	217
255	236	227	241	182
250	172	187	170	110
238	166	90	103	82
235	158	79	81	56

In what follows: image = vector $\quad \mathrm{x} \in \mathbb{R}^{N}$ with $N = N_1 N_2$

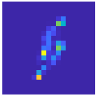
Basics about A: blur

$$\boxed{\mathrm{z} = \mathrm{A}\overline{\mathrm{x}} \quad \Leftrightarrow \quad \mathrm{z} = \phi * \overline{\mathrm{x}}}$$

Basics about A: blur

$$\boxed{\mathrm{z} = \mathrm{A}\overline{\mathrm{x}} \quad \Leftrightarrow \quad \mathrm{z} = \phi * \overline{\mathrm{x}}}$$

$=$



$*$

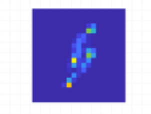
z ϕ $\overline{\mathrm{x}}$

Basics about A: blur

$$\boxed{\mathrm{z} = \mathrm{A}\overline{\mathrm{x}} \quad \Leftrightarrow \quad \mathrm{z} = \phi * \overline{\mathrm{x}}}$$

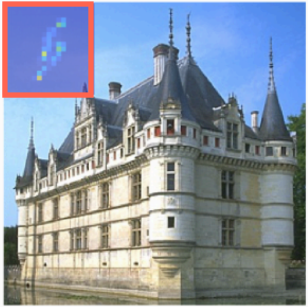
z

$=$



ϕ

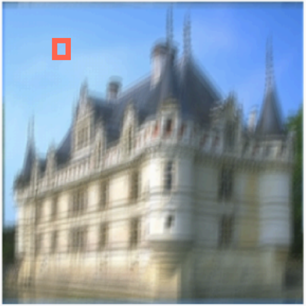
$*$



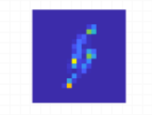
$\overline{\mathrm{x}}$

Basics about A: blur

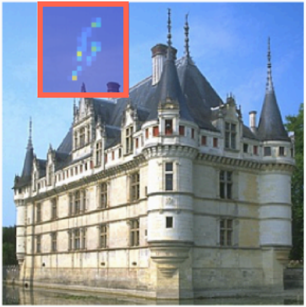
$$\boxed{\mathrm{z} = \mathrm{A}\overline{\mathrm{x}} \quad \Leftrightarrow \quad \mathrm{z} = \phi * \overline{\mathrm{x}}}$$



$=$



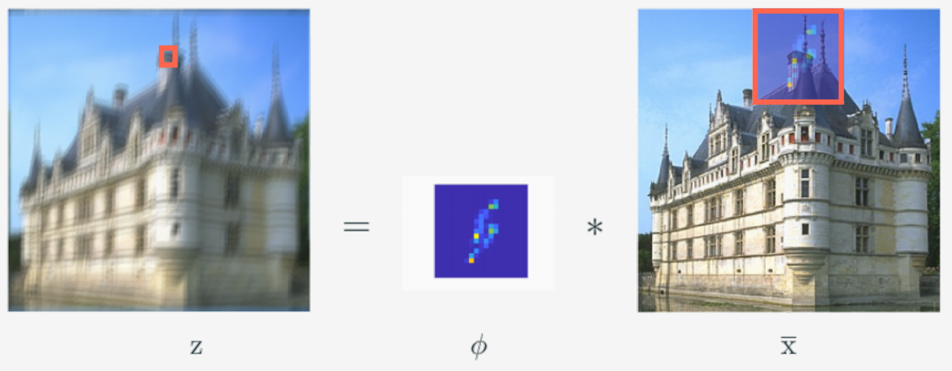
$*$



z $\qquad$ ϕ $\qquad$ $\overline{\mathrm{x}}$

Basics about A: blur

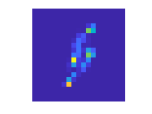
$$\boxed{\mathrm{z} = \mathrm{A}\overline{\mathrm{x}} \quad \Leftrightarrow \quad \mathrm{z} = \phi * \overline{\mathrm{x}}}$$



Basics about A: blur

$$\boxed{\mathrm{z} = \mathrm{A}\overline{\mathrm{x}} \quad \Leftrightarrow \quad \mathrm{z} = \phi * \overline{\mathrm{x}}}$$

$=$



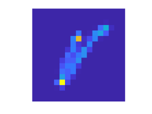
$*$

z ϕ $\overline{\mathrm{x}}$

Basics about A: blur

$$\boxed{\mathrm{z} = \mathrm{A}\overline{\mathrm{x}} \quad \Leftrightarrow \quad \mathrm{z} = \phi * \overline{\mathrm{x}}}$$

$=$



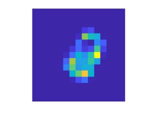
$*$

z $\qquad$ ϕ $\qquad$ $\overline{\mathrm{x}}$

Basics about A: blur

$$\boxed{\mathrm{z} = \mathrm{A}\overline{\mathrm{x}} \quad \Leftrightarrow \quad \mathrm{z} = \phi * \overline{\mathrm{x}}}$$

$=$



$*$

z ϕ $\overline{\mathrm{x}}$

Basics about A: high-pass filtering

$$\mathrm{z} = \mathrm{A}\overline{\mathrm{x}} \quad \Leftrightarrow \quad \mathrm{z} = \phi * \overline{\mathrm{x}}$$

$$= \quad [-1, 1] \quad *$$

z ϕ $\overline{\mathrm{x}}$

Basics about A: high-pass filtering

$$\mathrm{z} = \mathrm{A}\overline{\mathrm{x}} \quad \Leftrightarrow \quad \mathrm{z} = \phi * \overline{\mathrm{x}}$$

$$= \quad [-1, 1]^\top \quad *$$

z $\qquad$ ϕ $\qquad$ $\overline{\mathrm{x}}$

Hadamard conditions (1902)

The problem $\mathrm{z} = \mathrm{A}\overline{\mathrm{x}}$ is said to be well-posed if it fulfills the **Hadamard conditions** :

1. **existence of a solution,**

   i.e. the range A of A is equal to $\mathbb{R}^M$,

2. **uniqueness of the solution,**

   i.e. the nullspace $\ker \mathrm{A}$ of A is equal to $\{0\}$,

3. **stability of the solution $\widehat{\mathrm{x}}$ relatively to the observation,**

   i.e. $\left(\forall (\mathrm{z}, \mathrm{z}') \in \left(\mathbb{R}^M\right)^2\right)$

$$\|\mathrm{z} - \mathrm{z}'\| \to 0 \quad \Rightarrow \quad \|\widehat{\mathrm{x}}(\mathrm{z}) - \widehat{\mathrm{x}}(\mathrm{z}')\| \to 0.$$

Hadamard conditions (1902)

The problem $\mathrm{z} = \mathrm{A}\overline{\mathrm{x}}$ is said to be well-posed if it fulfills the **Hadamard conditions** :

1. **existence of a solution**,

   i.e. every vector z in $\mathbb{R}^M$ is the image of a vector x in $\mathbb{R}^N$,

2. **uniqueness of the solution**,

   i.e. if $\widehat{\mathrm{x}}(\mathrm{z})$ and $\widehat{\mathrm{x}}'(\mathrm{z})$ are two solutions, then they are necessarily equal since $\widehat{\mathrm{x}}(\mathrm{z}) - \widehat{\mathrm{x}}'(\mathrm{z})$ belongs to $\ker \mathrm{A}$,

3. **stability of the solution** $\widehat{\mathrm{x}}$ relatively to the observation,

   i.e. ensure that a small perturbation of the observed image leads to a slight variation of the recovered image.

Inverse model $\widehat{x} = d_\Theta(z)$ when $z = A\overline{x} + \varepsilon$

➡ [1922] **Maximum likelihood** (Fisher).

$$
\begin{aligned}
\widehat{x} \in \operatorname*{Argmin}_{x} \frac{1}{2}\|Ax - z\|_2^2 &= (A^*A)^{-1}A^*z \\
&= (A^*A)^{-1}A^*(A\overline{x} + \varepsilon) \\
&= \overline{x} + (A^*A)^{-1}A^*\varepsilon
\end{aligned}
$$

Inverse model $\widehat{x} = d_\Theta(z)$ when $z = A\overline{x} + \varepsilon$

➡ [1922] **Maximum likelihood** (Fisher).

$$\begin{aligned}
\widehat{x} \in \operatorname*{Argmin}_x \frac{1}{2}\|Ax - z\|_2^2 &= (A^*A)^{-1}A^*z \\
&= (A^*A)^{-1}A^*(A\overline{x} + \varepsilon) \\
&= \overline{x} + \boxed{(A^*A)^{-1}A^*\varepsilon}
\end{aligned}$$

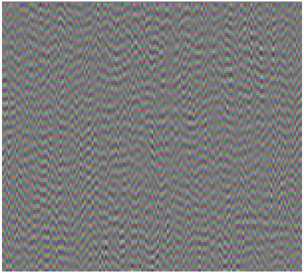
Noise amplification

Inverse model $\widehat{x} = d_\Theta(z)$ when $z = A\bar{x} + \varepsilon$

➡ [1922] **Maximum likelihood** (Fisher).

$$\begin{aligned}\widehat{x} \in \operatorname*{Argmin}_x \frac{1}{2}\|Ax - z\|_2^2 &= (A^*A)^{-1}A^*z \\ &= (A^*A)^{-1}A^*(A\bar{x} + \varepsilon) \\ &= \bar{x} + (A^*A)^{-1}A^*\varepsilon\end{aligned}$$

Noise amplification

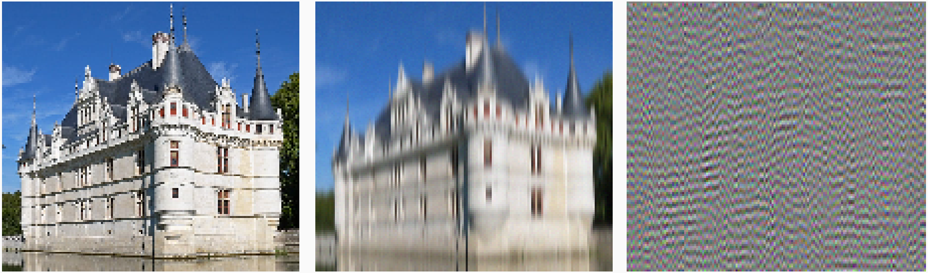


$\varepsilon \sim \mathcal{N}(0, 0.05 \times \mathrm{Id})$

Inverse model $\widehat{\mathrm{x}} = \mathrm{d}_\Theta(\mathrm{z})$ when $\mathrm{z} = \mathrm{A}\overline{\mathrm{x}} + \varepsilon$

→ [1922] **Maximum likelihood** (Fisher).

$$\begin{aligned}\widehat{\mathrm{x}} \in \operatorname*{Argmin}_{\mathrm{x}} \frac{1}{2}\|\mathrm{Ax} - \mathrm{z}\|_2^2 &= (\mathrm{A}^*\mathrm{A})^{-1}\mathrm{A}^*\mathrm{z} \\ &= (\mathrm{A}^*\mathrm{A})^{-1}\mathrm{A}^*(\mathrm{A}\overline{\mathrm{x}} + \varepsilon) \\ &= \overline{\mathrm{x}} + \underbrace{(\mathrm{A}^*\mathrm{A})^{-1}\mathrm{A}^*\varepsilon}_{\text{Noise amplification}}\end{aligned}$$



$\varepsilon \sim \mathcal{N}(0, 0.01 \times \mathrm{Id})$

Inverse model $\widehat{x} = d_\Theta(z)$ when $z = A\overline{x} + \varepsilon$

➡ [1922] **Maximum likelihood** (Fisher).

$$\widehat{x} \in \operatorname*{Argmin}_{x} \frac{1}{2}\|Ax - z\|_2^2 = (A^*A)^{-1}A^*z$$
$$= (A^*A)^{-1}A^*(A\overline{x} + \varepsilon)$$
$$= \overline{x} + (A^*A)^{-1}A^*\varepsilon$$

Noise amplification

$\varepsilon \sim \mathcal{N}(0, 0.001 \times \mathrm{Id})$

Inverse model $\widehat{x} = d_\Theta(z)$ when $z = A\bar{x} + \varepsilon$

➡ [1922] **Maximum likelihood** (Fisher).

$$\begin{aligned}\widehat{x} \in \operatorname*{Argmin}_x \frac{1}{2}\|Ax - z\|_2^2 &= (A^*A)^{-1}A^*z \\ &= (A^*A)^{-1}A^*(A\bar{x} + \varepsilon) \\ &= \bar{x} + \boxed{(A^*A)^{-1}A^*\varepsilon}\end{aligned}$$

Noise amplification

$\varepsilon \sim \mathcal{N}(0, 0 \times \mathrm{Id})$

Inverse model $\widehat{\mathrm{x}} = d_\Theta(\mathrm{z})$ when $\mathrm{z} = \mathrm{A}\overline{\mathrm{x}} + \varepsilon$

- [1922] **Maximum likelihood** (Fisher).

$$\widehat{\mathrm{x}} \in \operatorname*{Argmin}_{\mathrm{x}} \frac{1}{2}\|\mathrm{Ax} - \mathrm{z}\|_2^2 = (\mathrm{A}^*\mathrm{A})^{-1}\mathrm{A}^*\mathrm{z}$$

- [1963] **Regularisation** (Tikhonov, Huber)

$$\widehat{\mathrm{x}} \in \operatorname*{Argmin}_{\mathrm{x}} \frac{1}{2}\|\mathrm{Ax} - \mathrm{z}\|_2^2 + \theta\|\mathrm{Lx}\|_2^2 \qquad \text{avec} \quad \theta > 0$$

- [2000] **Sparsity** (Donoho, Daubechies-Defrise-DeMol,...)

$$\widehat{\mathrm{x}} \in \operatorname*{Argmin}_{\mathrm{x}} \frac{1}{2}\|\mathrm{Ax} - \mathrm{z}\|_2^2 + \theta\|\mathrm{Lx}\|_\star$$

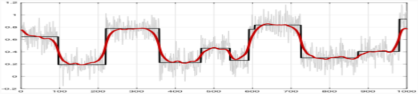
Regularized approaches

Piecewise constant denoising: $z = \overline{x} + \varepsilon$ with $\varepsilon = \mathcal{N}(0, \sigma^2 \mathrm{Id})$

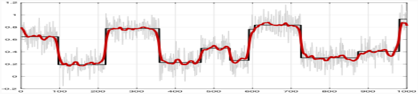
- **Minimisation problem**

$$\widehat{x}(z; \widehat{\theta}) = \arg\min_{x \in \mathbb{R}^N} \frac{1}{2}\|x - z\|_2^2 + \theta \, \|Lx\|_{\bullet} \quad \text{where} \quad \begin{cases} Lx = \psi * x \\ \theta > 0 \end{cases}$$

- **Linear denoising**



$\psi = \begin{bmatrix} 1 & -1 \end{bmatrix}$; $\| \cdot \|_{\bullet} = \| \cdot \|_2^2$; Large θ



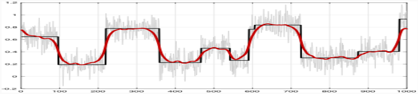
$\psi = \begin{bmatrix} 1 & -1 \end{bmatrix}$; $\| \cdot \|_{\bullet} = \| \cdot \|_2^2$; Small θ

Piecewise constant denoising: $\mathrm{z} = \overline{\mathrm{x}} + \varepsilon$ with $\varepsilon = \mathcal{N}(0, \sigma^2 \mathrm{Id})$

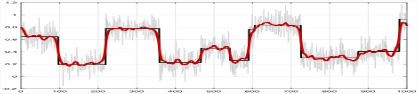
- **Minimisation problem**

$$\widehat{\mathrm{x}}(\mathrm{z}; \widehat{\theta}) = \arg\min_{\mathrm{x} \in \mathbb{R}^N} \frac{1}{2}\|\mathrm{x} - \mathrm{z}\|_2^2 + \theta \, \|\mathrm{Lx}\|_{\bullet} \quad \text{where} \quad \begin{cases} \mathrm{Lx} = \psi * \mathrm{x} \\ \theta > 0 \end{cases}$$

- **Linear denoising**

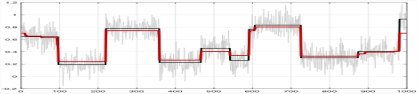


$\psi = \begin{bmatrix} 1 & -1 \end{bmatrix}$; $\| \cdot \|_{\bullet} = \| \cdot \|_2^2$; Large θ

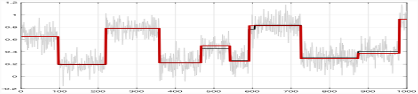


$\psi = \begin{bmatrix} 1 & -1 \end{bmatrix}$; $\| \cdot \|_{\bullet} = \| \cdot \|_2^2$; Small θ

- **Nonlinear denoising**



$\mathrm{L} = \begin{bmatrix} 1 & -1 \end{bmatrix}$ and $\| \cdot \|_{\bullet} = \| \cdot \|_1$



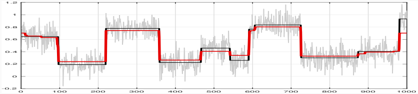
$\psi = \begin{bmatrix} 1 & -1 \end{bmatrix}$ and $\| \cdot \|_{\bullet} = \| \cdot \|_0$

Piecewise constant denoising: $\mathrm{z} = \overline{\mathrm{x}} + \varepsilon$ avec $\varepsilon = \mathcal{N}(0, \sigma^2 \mathrm{Id})$

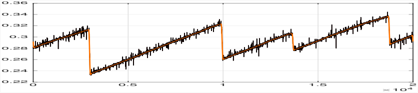
- **Minimisation problem**

$$\widehat{\mathrm{x}}(\mathrm{z}; \widehat{\theta}) = \arg \min_{\mathrm{x} \in \mathbb{R}^N} \frac{1}{2} \|\mathrm{x} - \mathrm{z}\|_2^2 + \theta \, \|\mathrm{Lx}\|_\bullet \quad \text{where} \quad \begin{cases} \mathrm{Lx} = \psi * \mathrm{x} \\ \theta > 0 \end{cases}$$

- **Nonlinear denoising: piecewise constant/linear**



$\psi = \begin{bmatrix} 1 & -1 \end{bmatrix}$ and $\| \cdot \|_\bullet = \| \cdot \|_1$

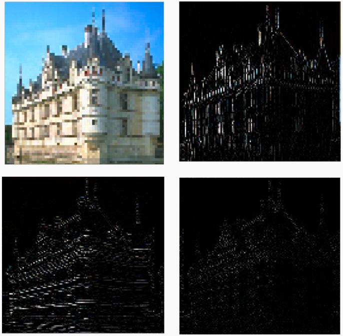


$\psi = \begin{bmatrix} 1 & -2 & 1 \end{bmatrix}$ and $\| \cdot \|_\bullet = \| \cdot \|_1$

Wavelets denoising: $z = \bar{x} + \varepsilon$ with $\varepsilon = \mathcal{N}(0, \sigma^2 \mathrm{Id})$

1 level wavelet decomposition

x

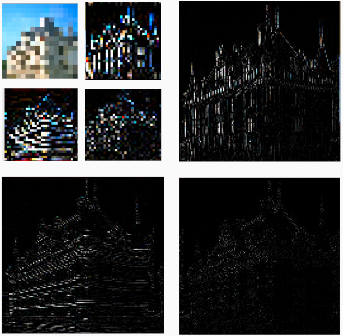


$\zeta = Lx$

Wavelets denoising: $\mathrm{z} = \overline{\mathrm{x}} + \varepsilon$ with $\varepsilon = \mathcal{N}(0, \sigma^2 \mathrm{Id})$

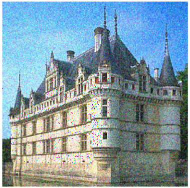
2 levels wavelet decomposition

x

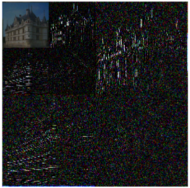


$\zeta = \mathrm{Lx}$

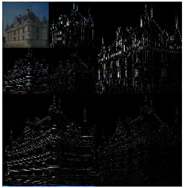
Wavelets denoising: $\mathrm{z} = \overline{\mathrm{x}} + \varepsilon$ with $\varepsilon = \mathcal{N}(0, \sigma^2 \mathrm{Id})$



z



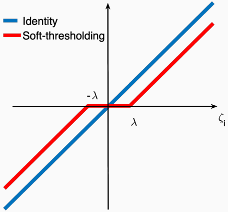
$\boldsymbol{\zeta} = \mathcal{W}\mathrm{z}$



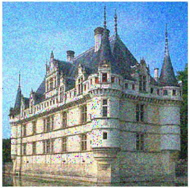
$\mathrm{soft}_\theta(\mathcal{W}\mathrm{z})$

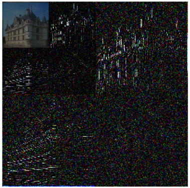
$\widehat{\mathrm{x}} = \mathcal{W}^* \mathrm{soft}_\theta(\mathcal{W}\mathrm{z})$

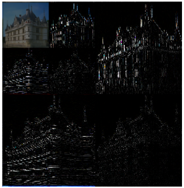
$$\begin{aligned}\mathrm{soft}_\theta(\boldsymbol{\zeta}) &= \big(\max\{|\zeta_{\underline{i}}| - \theta, 0\}\mathrm{sign}(\zeta_{\underline{i}})\big)_{\underline{i}\in\Omega} \\ &= \arg\min_{\boldsymbol{\nu}} \frac{1}{2}\|\boldsymbol{\nu} - \boldsymbol{\zeta}\|_2^2 + \theta\|\boldsymbol{\nu}\|_1\end{aligned}$$



Wavelets denoising: $\mathrm{z} = \overline{\mathrm{x}} + \varepsilon$ with $\varepsilon = \mathcal{N}(0, \sigma^2 \mathrm{Id})$

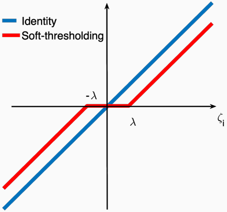




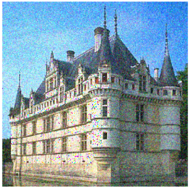


z | $\zeta = \mathcal{W}\mathrm{z}$ | $\mathrm{soft}_\theta(\mathcal{W}\mathrm{z})$ | $\widehat{\mathrm{x}} = \mathcal{W}^* \mathrm{soft}_\theta(\mathcal{W}\mathrm{z})$

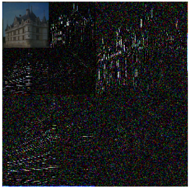
$$\begin{aligned}\mathrm{soft}_\theta(\boldsymbol{\zeta}) &= \big(\max\{|\zeta_{\underline{i}}| - \theta, 0\}\mathrm{sign}(\zeta_{\underline{i}})\big)_{\underline{i}\in\Omega} \\ &= \arg\min_{\boldsymbol{\nu}} \frac{1}{2}\|\boldsymbol{\nu} - \boldsymbol{\zeta}\|_2^2 + \theta\|\boldsymbol{\nu}\|_1\end{aligned}$$



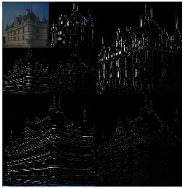
Wavelets denoising: $\mathrm{z} = \overline{\mathrm{x}} + \varepsilon$ with $\varepsilon = \mathcal{N}(0, \sigma^2 \mathrm{Id})$



z



$\boldsymbol{\zeta} = \mathcal{W}\mathrm{z}$



$\mathrm{soft}_\theta(\mathcal{W}\mathrm{z})$

$\widehat{\mathrm{x}} = \mathcal{W}^* \mathrm{soft}_\theta(\mathcal{W}\mathrm{z})$

$$\begin{aligned} \mathrm{soft}_\theta(\boldsymbol{\zeta}) &= \big(\max\{|\zeta_{\underline{i}}| - \theta, 0\} \mathrm{sign}(\zeta_{\underline{i}}) \big)_{\underline{i} \in \Omega} \\ &= \arg\min_{\boldsymbol{\nu}} \frac{1}{2} \|\boldsymbol{\nu} - \boldsymbol{\zeta}\|_2^2 + \theta \|\boldsymbol{\nu}\|_1 \end{aligned}$$

$$\begin{aligned} \widehat{\mathrm{x}} &= \mathcal{W}^* \mathrm{soft}_\theta(\mathcal{W}\mathrm{z}) \\ &= \arg\min_{\mathbf{x}} \frac{1}{2} \|\mathbf{x} - \mathbf{z}\|_2^2 + \theta \|\mathcal{W}\mathrm{x}\|_1 \end{aligned}$$

Inverse model $\widehat{x} = d_\Theta(z)$ with $z = A\overline{x} + \varepsilon$

$$\widehat{x} \in \mathrm{Argmin}_x \; \frac{1}{2}\|Ax - z\|_2^2 + \lambda \|Lx\|_*$$

MAP Bayesian interpretation $\Big\downarrow$ (likelihood / prior)

$$\widehat{x} \in \mathrm{Argmin}_{\underbrace{x \in C}_{\text{Constraint}}} \; \underbrace{f_1(Ax, z)}_{\text{Data fidelity}} + \lambda \underbrace{f_2(Lx)}_{\text{Regularisation}}$$

Inverse model $\widehat{\mathrm{x}} = \mathrm{d}_\Theta(\mathrm{z})$ with $\mathrm{z} = \mathrm{A}\overline{\mathrm{x}} + \varepsilon$

$$\widehat{\mathrm{x}} \in \mathrm{Argmin}_{\mathrm{x}} \ \frac{1}{2}\|\mathrm{Ax} - \mathrm{z}\|_2^2 + \lambda\|\mathrm{Lx}\|_*$$

MAP Bayesian interpretation $\downarrow$ (likelihood / prior)

$$\widehat{\mathrm{x}} \in \mathrm{Argmin}_{\mathrm{x}\in C} \ f_1(\mathrm{Ax}, \mathrm{z}) + \lambda f_2(\mathrm{Lx})$$

Minimisation algorithm

$$\mathrm{x}^{[k+1]} = \mathbf{T}(\mathrm{x}^{[k]})$$

Hyperparameter(s) estimation:
SURE, GCV, MCMC,...

Maximum A Posteriori (MAP)

Maximum A Posteriori (MAP)

Let x and z be random vector realizations X and Z.

$$\widehat{\mathrm{x}} \in \operatorname*{Argmax}_{\mathrm{x}\in\mathbb{R}^N} \quad \underbrace{\mu_{X|Z=\mathrm{z}}(\mathrm{x})}_{\text{Posterior distribution}}$$

Maximum A Posteriori (MAP)

Maximum A Posteriori (MAP)

Let x and z be random vector realizations X and Z.

$$\widehat{\mathrm{x}} \in \operatorname*{Argmax}_{\mathrm{x}\in\mathbb{R}^N} \quad \underbrace{\mu_{X|Z=\mathrm{z}}(\mathrm{x})}_{\text{Posterior distribution}}$$

Bayes rule:

$$\max_{\mathrm{x}\in\mathbb{R}^N} \mu_{X|Z=z}(\mathrm{x}) \Leftrightarrow \max_{\mathrm{x}\in\mathbb{R}^N} \mu_{Z|X=\mathrm{x}}(\mathrm{z}) \cdot \mu_X(\mathrm{x})$$

$$\Leftrightarrow \min_{\mathrm{x}\in\mathbb{R}^N} \left\{ -\log(\mu_{Z|X=\mathrm{x}}(\mathrm{z})) - \log(\mu_X(\mathrm{x})) \right\}$$

Maximum A Posteriori (MAP)

Maximum A Posteriori (MAP)

Let x and z be random vector realizations X and Z.

$$\widehat{\mathrm{x}} \in \underset{\mathrm{x}\in\mathbb{R}^N}{\operatorname{Argmax}} \quad \underbrace{\mu_{X|Z=\mathrm{z}}(\mathrm{x})}_{\text{Posterior distribution}}$$

Bayes rule:

$$\begin{aligned}
\max_{\mathrm{x}\in\mathbb{R}^N} \mu_{X|Z=z}(\mathrm{x}) &\Leftrightarrow \max_{\mathrm{x}\in\mathbb{R}^N} \mu_{Z|X=\mathrm{x}}(\mathrm{z}) \cdot \mu_X(\mathrm{x}) \\
&\Leftrightarrow \min_{\mathrm{x}\in\mathbb{R}^N} \Big\{ \underbrace{-\log(\mu_{Z|X=\mathrm{x}}(\mathrm{z}))}_{\text{Data-term}} \underbrace{-\log(\mu_X(\mathrm{x}))}_{\text{A priori}} \Big\} \\
&\Leftrightarrow \min_{\mathrm{x}\in\mathbb{R}^N} f_1(\mathrm{x}) + f_2(\mathrm{x})
\end{aligned}$$

Data-fidelity term: Gaussian noise

$$(\forall \mathrm{x} \in \mathbb{R}^N) \quad \color{red}{f_1}(\mathrm{x}) = -\log(\mu_{Z|X=\mathrm{x}}(\mathrm{z}))$$

- Let $\mathrm{z} = \mathrm{L}\overline{\mathrm{x}} + \varepsilon$ with $\varepsilon \sim \mathcal{N}(0, \alpha \mathrm{Id})$ with $\alpha > 0$.
- Gaussian likelihood:

$$\mu_{Z|X=\mathrm{x}}(\mathrm{z}) = \prod_{n=1}^{M} \frac{1}{\sqrt{2\pi\alpha}} \exp\left(\frac{((\mathrm{Lx})_n - \mathrm{z}_n)^2}{2\alpha} \right)$$

- Data-term:

$$f_1(\mathrm{x}) = \sum_{n=1}^{M} \frac{1}{2\alpha} ((\mathrm{Lx})_n - \mathrm{z}_n)^2$$

Data-fidelity term: Poisson noise

$$\boxed{(\forall \mathrm{x} \in \mathbb{R}^N) \quad f_1(\mathrm{x}) = -\log(\mu_{Z|X=\mathrm{x}}(\mathrm{z}))}$$

- Let $\mathrm{z} = \mathcal{D}_\alpha(\mathrm{L}\overline{\mathrm{x}})$ where $\mathcal{D}_\alpha$ Poisson noise with parameter α.
- Poisson likelihood:

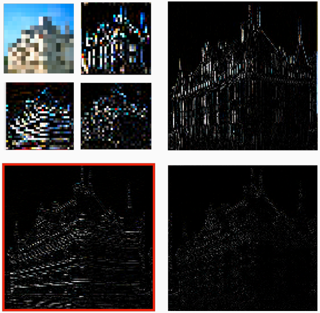
$$\mu_{Z|X=\mathrm{x}}(\mathrm{z}) = \prod_{n=1}^{M} \frac{\exp\big(-\alpha(\mathrm{Lx})_n\big)}{z_n!} \big(\alpha(\mathrm{Lx})_n\big)^{\mathrm{z}_n}$$

- Data-term: $f_1(\mathrm{x}) = \sum_{n=1}^{M} \Psi_i\big((\mathrm{Lx})_n\big)$

$$(\forall \upsilon \in \mathbb{R}) \quad \Psi_i(\upsilon) = \begin{cases} \alpha\upsilon - \mathrm{z}_n \log(\alpha\upsilon) & \text{if } \mathrm{z}_n > 0 \text{ and } \upsilon > 0, \\ \alpha\upsilon & \text{si } \mathrm{z}_n = 0 \text{ and } \upsilon \geq 0, \\ +\infty & \text{otherwise.} \end{cases}$$

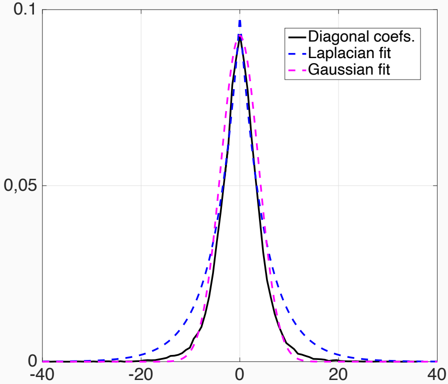
Bayesian interpretation

$$(\forall \mathrm{x} \in \mathbb{R}^N) \quad f_2(\mathrm{x}) = -\log(\mu_X(\mathrm{x})))$$



$$\zeta = \mathrm{Lx}$$

Wavelet coefficients



Probability density function

Context: image restoration

Synthesis formulation

$\widehat{\mathrm{x}} = \mathrm{L}^* \widehat{\zeta}$ with $\mathrm{L} \in \mathbb{R}^{P \times N}$

$$\widehat{\zeta} \in \underset{\zeta}{\operatorname{Argmin}} \frac{1}{2} \|\mathrm{A} \mathrm{L}^* \zeta - \mathrm{z}\|_2^2 + \lambda \|\zeta\|_\bullet$$

Analysis formulation

$$\widehat{\mathrm{x}} \in \underset{\mathrm{x}}{\operatorname{Argmin}} \frac{1}{2} \|\mathrm{A}\mathrm{x} - \mathrm{z}\|_2^2 + \lambda \|\mathrm{L}\mathrm{x}\|_\bullet$$

$\Rightarrow$ **Equivalence for L orthonormal basis.**

[Elad, Milanfar, Ron, 2007; Chaari, Pustelnik, Chaux, Pesquet, 2009; Selesnick, Figueiredo, 2009; Carlavan, Weiss, Blanc-Féraud, 2010; Pustelnik, Benazza-Benhayia, Zheng, Pesquet, 2010.

Context: image restoration

Synthesis formulation

$\widehat{\mathrm{x}} = \mathrm{L}^* \widehat{\zeta}$ with $\mathrm{L} \in \mathbb{R}^{P \times N}$

$$\widehat{\zeta} \in \underset{\zeta}{\mathrm{Argmin}} \frac{1}{2} \| \mathrm{A} \mathrm{L}^* \zeta - \mathrm{z} \|_2^2 + \lambda \| \zeta \|_\bullet$$

Analysis formulation

$$\widehat{\mathrm{x}} \in \underset{\mathrm{x}}{\mathrm{Argmin}} \frac{1}{2} \| \mathrm{A} \mathrm{x} - \mathrm{z} \|_2^2 + \lambda \| \mathrm{L} \mathrm{x} \|_\bullet$$

$\Rightarrow$ **Equivalence for L orthonormal basis.**

- X-lets
- Sparse coding

- Horizontal/vertical gradients: TV
- Hessian operator
- Nonlocal total variation: weighted nonlocal gradients: NLTV
- Local dictionaries of patches

[webpage L. Duval; Aharon, Elad, Bruckstein, 2006; Mairal, Sapiro, Elad, 2007; Gilboa, Osher, 2008; K Bredies, K Kunisch, T Pock, 2010; Jacques, Duval, Chaux, Peyré, 2011; S Lefkimmiatis, A Bourquard, M Unser,2011; Zoran, Weiss, 2011; G Kutyniok, D Labate, 2012; Chierchia et al.,2014; Boulanger et al., 2018;. . .]

Subdifferential, Proximity operator, Proximal algorithms

From smooth to non-smooth optimization to solve $\widehat{\mathbf{x}} \in \underset{\mathbf{x}}{\mathrm{Argmin}}\ f(\mathbf{x})$

Gradient descent

$$\mathrm{x}^{[k+1]} = \mathrm{x}^{[k]} - \tau \nabla f(\mathrm{x}^{[k]})$$

(Forward) subgradient descent

$$\mathrm{x}^{[k+1]} = \mathrm{x}^{[k]} - \mathrm{u}^{[k]} \quad \text{where} \quad \mathrm{u}^{[k]} \in \tau \partial f(\mathrm{x}^{[k]})$$

(Backward) subgradient descent

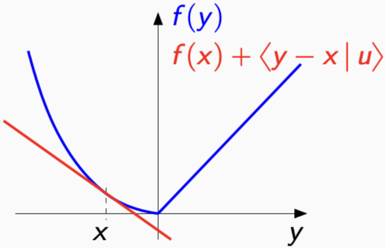
$$\mathrm{x}^{[k+1]} = \mathrm{x}^{[k]} - \mathrm{u}^{[k]} \quad \text{where} \quad \mathrm{u}^{[k]} \in \tau\, \partial f(\mathrm{x}^{[k+1]})$$

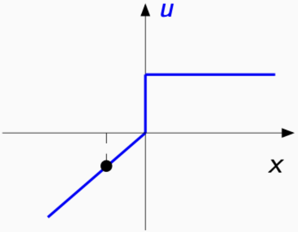
Subdifferential

Let $f : \mathcal{H} \to$ be a proper function.

The **Moreau subdifferential of f**, denoted ∂f, is such that:

$$\partial f : \mathcal{H} \to 2^{\mathcal{H}}$$
$$\mathrm{x} \to \{\mathrm{u} \in \mathcal{H} \,|\, (\forall \mathrm{y} \in \mathcal{H})\ \langle \mathrm{y} - \mathrm{x} | \mathrm{u} \rangle + f(\mathrm{x}) \leq f(\mathrm{y})\}$$





Subdifferential

Let $f : \mathcal{H} \to$ be a proper function.

The **Moreau subdifferential of f**, denoted ∂f, is such that:

$$\begin{aligned} \partial f : \mathcal{H} &\to 2^{\mathcal{H}} \\ \mathrm{x} &\to \{\mathrm{u} \in \mathcal{H} \,|\, (\forall \mathrm{y} \in \mathcal{H})\ \langle \mathrm{y} - \mathrm{x} | \mathrm{u} \rangle + f(\mathrm{x}) \le f(\mathrm{y})\} \end{aligned}$$

Fermat rule: $\quad 0 \in \partial f(\widehat{\mathrm{x}}) \quad \Leftrightarrow \quad \widehat{\mathrm{x}} \in \underset{\mathrm{x}}{\mathrm{Argmin}}\ f(\mathrm{x})$

Forward subgradient descent

$$\mathrm{x}^{[k+1]} = \mathrm{x}^{[k]} - \mathrm{u}^{[k]} \ \text{ where } \ \mathrm{u}^{[k]} \in \tau \partial f(\mathrm{x}^{[k]})$$

Proximity operator

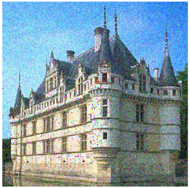
Definition [Moreau,1965] Let $f : \mathcal{H} \to]-\infty, +\infty]$ be a proper, l.s.c, and convex function. The proximity operator of f at point $\mathrm{x} \in \mathcal{H}$ is the **unique point** denoted $\mathrm{prox}_f \mathrm{x}$ such that

$$(\forall \mathrm{x} \in \mathcal{H}) \qquad \mathrm{prox}_f \mathrm{x} = \arg\min_{\mathrm{v} \in \mathcal{H}} \frac{1}{2} \|\mathrm{x} - \mathrm{v}\|^2 + f(\mathrm{v})$$

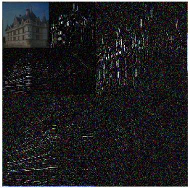
➡ **Numerous closed forms are available**

- $\mathrm{prox}_{\iota_C} = \mathrm{P}_C$: **Projection** onto a convex set.
- $\mathrm{prox}_{\theta\|\cdot\|_1}$: **soft-thresholding** with threshold $\theta > 0$.
- Full list available: **PROX Repository**

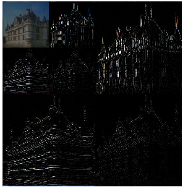
Wavelets denoising: $z = \bar{x} + \varepsilon$ with $\varepsilon = \mathcal{N}(0, \sigma^2 \mathrm{Id})$



z



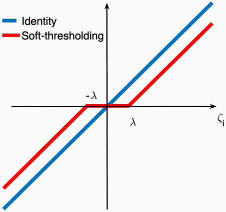
$\zeta = \mathcal{W}z$



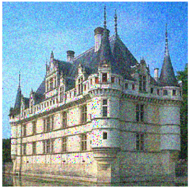
$\mathrm{soft}_\theta(\mathcal{W}z)$

$\widehat{x} = \mathcal{W}^* \mathrm{soft}_\theta(\mathcal{W}z)$

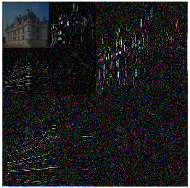
$$
\begin{aligned}
\mathrm{soft}_\theta(\boldsymbol{\zeta}) &= \big(\max\{|\zeta_{\underline{i}}| - \theta, 0\} \mathrm{sign}(\zeta_{\underline{i}}) \big)_{\underline{i} \in \Omega} \\
&= \arg\min_{\boldsymbol{\nu}} \frac{1}{2} \|\boldsymbol{\nu} - \boldsymbol{\zeta}\|_2^2 + \theta \|\boldsymbol{\nu}\|_1
\end{aligned}
$$



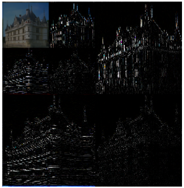
Wavelets denoising: $z = \bar{x} + \varepsilon$ with $\varepsilon = \mathcal{N}(0, \sigma^2 \mathrm{Id})$



z



$\boldsymbol{\zeta} = \mathcal{W}z$

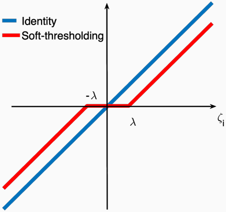


$\mathrm{soft}_\theta(\mathcal{W}z)$

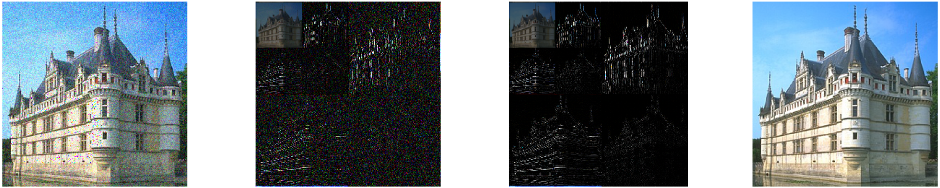
$\widehat{x} = \mathcal{W}^* \mathrm{soft}_\theta(\mathcal{W}z)$

$$
\begin{aligned}
\mathrm{soft}_\theta(\boldsymbol{\zeta}) &= \big(\max\{|\zeta_{\underline{i}}| - \theta, 0\}\mathrm{sign}(\zeta_{\underline{i}})\big)_{\underline{i}\in\Omega} \\
&= \arg\min_{\boldsymbol{\nu}} \frac{1}{2}\|\boldsymbol{\nu} - \boldsymbol{\zeta}\|_2^2 + \theta\|\boldsymbol{\nu}\|_1 \\
&= \mathrm{prox}_{\theta\|\cdot\|_1}(\boldsymbol{\zeta})
\end{aligned}
$$

$\rightarrow$ **proximity operator**



Wavelets denoising: $\mathrm{z} = \bar{\mathrm{x}} + \varepsilon$ with $\varepsilon = \mathcal{N}(0, \sigma^2 \mathrm{Id})$



z $\qquad$ $\boldsymbol{\zeta} = \mathcal{W}\mathrm{z}$ $\qquad$ $\mathrm{soft}_\theta(\mathcal{W}\mathrm{z})$ $\qquad$ $\widehat{\mathrm{x}} = \mathcal{W}^* \mathrm{soft}_\theta(\mathcal{W}\mathrm{z})$

$$
\begin{aligned}
\mathrm{soft}_\theta(\boldsymbol{\zeta}) &= \Big(\max\{|\zeta_{\underline{i}}| - \theta, 0\} \mathrm{sign}(\zeta_{\underline{i}}) \Big)_{\underline{i} \in \Omega} \\
&= \arg\min_{\boldsymbol{\nu}} \frac{1}{2} \|\boldsymbol{\nu} - \boldsymbol{\zeta}\|_2^2 + \theta \|\boldsymbol{\nu}\|_1 \\
&= \boxed{\mathrm{prox}_{\theta\|\cdot\|_1}}(\boldsymbol{\zeta}) \quad \to \textbf{proximity operator}
\end{aligned}
$$

$$
\begin{aligned}
\widehat{\mathrm{x}} &= \mathcal{W}^* \mathrm{soft}_\theta(\mathcal{W}\mathrm{z}) \\
&= \arg\min_{\mathbf{x}} \frac{1}{2} \|\mathbf{x} - \mathbf{z}\|_2^2 + \theta \|\mathcal{W}\mathrm{x}\|_1 = \boxed{\mathrm{prox}_{\theta\|\mathcal{W}\cdot\|_1}(\mathrm{z})}
\end{aligned}
$$

Proximity operator

Let $\mathcal{H}$ be a Hilbert space and $f \in \Gamma_0(\mathcal{H})$.

$$(\forall \mathrm{x} \in \mathcal{H}) \quad \mathrm{p} = \mathrm{prox}_f(\mathrm{x}) \quad \Leftrightarrow \quad \mathrm{x} - \mathrm{p} \in \partial f(\mathrm{p}).$$

Proximity operator

Let $\mathcal{H}$ be a Hilbert space and $f \in \Gamma_0(\mathcal{H})$.

$$(\forall \mathrm{x} \in \mathcal{H}) \quad \mathrm{p} = \mathrm{prox}_f(\mathrm{x}) \quad \Leftrightarrow \quad \mathrm{x} - \mathrm{p} \in \partial f(\mathrm{p}).$$

- Proof:

$$\begin{aligned} \mathrm{p} = \underset{\mathrm{y} \in \mathcal{H}}{\arg\min}\ f(\mathrm{y}) + \frac{1}{2}\|\mathrm{y} - \mathrm{x}\|^2 \quad &\Leftrightarrow \quad 0 \in \partial\Big(f + \frac{1}{2}\|\cdot - \mathrm{x}\|^2\Big)(\mathrm{p}) \\ &\Leftrightarrow \quad 0 \in \partial f(\mathrm{p}) + \mathrm{p} - \mathrm{x} \end{aligned}$$

Proximity operator

Let $\mathcal{H}$ be a Hilbert space and $f \in \Gamma_0(\mathcal{H})$.

$$(\forall \mathrm{x} \in \mathcal{H}) \quad \mathrm{p} = \mathrm{prox}_f(\mathrm{x}) \quad \Leftrightarrow \quad \mathrm{x} - \mathrm{p} \in \partial f(\mathrm{p}).$$

- Proof:

$$\mathrm{p} = \underset{\mathrm{y} \in \mathcal{H}}{\arg\min} \; f(\mathrm{y}) + \frac{1}{2}\|\mathrm{y} - \mathrm{x}\|^2 \quad \Leftrightarrow \quad 0 \in \partial\Big(f + \frac{1}{2}\|\cdot - \mathrm{x}\|^2\Big)(\mathrm{p})$$

$$\Leftrightarrow \quad 0 \in \partial f(\mathrm{p}) + \mathrm{p} - \mathrm{x}$$

Proximity operator

Let $\mathcal{H}$ be a Hilbert space and $f \in \Gamma_0(\mathcal{H})$.

$$(\forall \mathrm{x} \in \mathcal{H}) \quad \mathrm{p} = \mathrm{prox}_f(\mathrm{x}) \quad \Leftrightarrow \quad \mathrm{x} - \mathrm{p} \in \partial f(\mathrm{p}).$$

Backward subgradient descent $=$ **Proximal point algorithm**

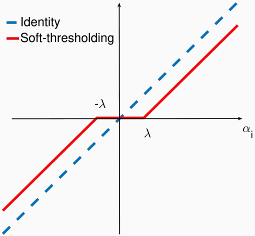
$$\begin{aligned} \mathrm{x}^{[k+1]} &= \mathrm{x}^{[k]} - \mathrm{u}^{[k]} \ \text{ where } \ \mathrm{u}^{[k]} \in \tau \partial f(\mathrm{x}^{[k+1]}) \\ &= \mathrm{prox}_{\tau f}(\mathrm{x}^{[k]}) \end{aligned}$$

Proximal algorithm to solve $\widehat{\mathbf{x}} \in \underset{\mathbf{x}}{\mathrm{Argmin}}\ f(\mathbf{x})$

Key tool – *Proximity operator*

Backward subgradient descent:
$$\begin{aligned} \mathrm{x}^{[k+1]} &= \mathrm{x}^{[k]} - \mathrm{u}^{[k]} \text{ where } \mathrm{u}^{[k]} \in \tau \partial f(\mathrm{x}^{[k+1]}) \\ &= \mathrm{prox}_{\tau f}(\mathrm{x}^{[k]}) \end{aligned}$$

➡ Proximity operator of the ℓ_1 norm = soft-thresholding



Proximal algorithm to solve $\widehat{\mathrm{x}} \in \underset{\mathbf{x}}{\mathrm{Argmin}}\ f(\mathbf{x})$

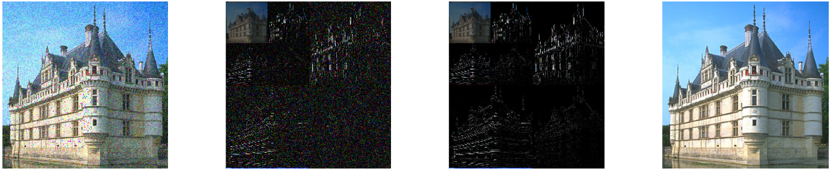
Key tool – *Proximity operator*

Backward subgradient descent: $\mathrm{x}^{[k+1]} = \mathrm{x}^{[k]} - \mathrm{u}^{[k]}$ where $\mathrm{u}^{[k]} \in \tau\partial f(\mathrm{x}^{[k+1]})$

$$= \mathrm{prox}_{\tau f}(\mathrm{x}^{[k]})$$

➡ Proximity operator of the ℓ_1 norm $=$ soft-thresholding

➡ Denoising (L orthon. basis): $\widehat{\mathrm{x}} = \mathrm{prox}_{\tau\theta\|\mathrm{L}\cdot\|_1}(\mathrm{z}) = \mathrm{L}^*\mathrm{soft}_{\gamma\theta}(\mathrm{Lz})$



z $\qquad$ $\mathrm{u} = \mathrm{Lz}$ $\qquad$ $\mathrm{soft}_{\tau\theta}(\mathrm{Lz})$ $\qquad$ $\widehat{\mathrm{x}} = \mathrm{L}^*\mathrm{soft}_{\theta}(\mathrm{Lz})$

Proximal algorithm to solve $\widehat{\mathrm{x}} \in \underset{\mathbf{x}}{\mathrm{Argmin}}\ f(\mathbf{x})$

Key tool – *Proximity operator*

Backward subgradient descent:
$$\begin{aligned} \mathrm{x}^{[k+1]} &= \mathrm{x}^{[k]} - \mathrm{u}^{[k]} \text{ where } \mathrm{u}^{[k]} \in \tau\partial f(\mathrm{x}^{[k+1]}) \\ &= \mathrm{prox}_{\tau f}(\mathrm{x}^{[k]}) \end{aligned}$$

➡ Proximity operator of the ℓ_1 norm $=$ soft-thresholding

➡ Denoising (L orthon. basis): $\widehat{\mathrm{x}} = \mathrm{prox}_{\tau\theta\|\mathrm{L}\cdot\|_1}(\mathrm{z}) = \mathrm{L}^*\mathrm{soft}_{\gamma\theta}(\mathrm{Lz})$

➡ Numerous closed forms: **http://proximity-operator.net**.

Proximity operator: examples

Projection:

Let $\mathcal{H}$ be a Hilbert space. Let C be a nonempty closed convex subset of $\mathcal{H}$.

$$(\forall x \in \mathcal{H}) \qquad \operatorname{prox}_{\iota_C}(x) = \underset{y \in C}{\operatorname{argmin}} \frac{1}{2}\|y - x\|^2 = P_C(x).$$

Proximity operator: examples

Power q function with $q \geq 1$:

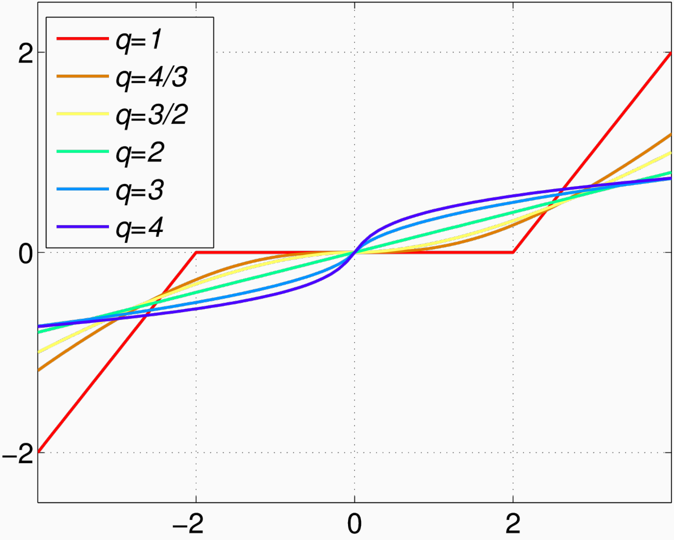
Let $\theta > 0$, $q \in [1, +\infty[$ and $f\colon \mathbb{R} \to : \eta \mapsto \theta|\xi|^q$.

Then, for every $\xi \in \mathbb{R}$,

$$\operatorname{prox}_f \xi = \begin{cases} \boxed{\operatorname{sign}(\xi)\max\{|\xi| - \theta, 0\}} & \text{if } q = 1 \\ \xi + \frac{4\theta}{3\,.\,2^{1/3}}\left((\epsilon - \xi)^{1/3} - (\epsilon + \xi)^{1/3}\right) & \\ \qquad \text{where} \quad \epsilon = \sqrt{\xi^2 + 256\theta^3/729} & \text{if } q = \frac{4}{3} \\ \xi + \frac{9\theta^2 \operatorname{sign}(\xi)}{8}\left(1 - \sqrt{1 + \frac{16|\xi|}{9\theta^2}}\right) & \text{if } q = \frac{3}{2} \\ \frac{\xi}{1+2\theta} & \text{if } q = 2 \\ \operatorname{sign}(\xi)\frac{\sqrt{1+12\theta|\xi|}-1}{6\theta} & \text{if } q = 3 \\ \left(\frac{\epsilon+\xi}{8\theta}\right)^{1/3} - \left(\frac{\epsilon-\xi}{8\theta}\right)^{1/3} \quad \text{where} \quad \epsilon = \sqrt{\xi^2 + 1/(27\theta)} & \text{if } q = 4 \end{cases}$$

Proximity operator: examples

Power q function with $q \geq 1$ and $\theta = 2$.



Proximity operator: examples

Quadratic function:

Let $\mathrm{A} \in \mathbb{R}^{M \times N}$, $\tau > 0$ and $z \in \mathcal{G}$.

$$f = \tau \left\| \mathrm{A} \cdot - z \right\|^2 / 2 \quad \Rightarrow \quad \mathrm{prox}_f = (\mathrm{Id} + \tau \mathrm{A}^* \mathrm{A})^{-1} (\cdot + \tau \mathrm{A}^* z).$$

Proximity operator: properties

Let $\mathcal{H}$ and $\mathcal{G}$ be two Hilbert spaces. Let $f \in \Gamma_0(\mathcal{H})$ and $\mathrm{L} \in \mathcal{B}(\mathcal{G}, \mathcal{H})$ such that $\mathrm{L}\mathrm{L}^* = \mu \mathrm{Id}$ where $\mu > 0$. Then

$$\mathrm{prox}_{f \circ \mathrm{L}} = \mathrm{Id} - \mu^{-1} \mathrm{L}^* \circ (\mathrm{Id} - \mathrm{prox}_{\mu f}) \circ \mathrm{L}.$$

Proximity operator: properties

Let $\mathcal{H}$ and $\mathcal{G}$ be two Hilbert spaces. Let $f \in \Gamma_0(\mathcal{H})$ and $\mathrm{L} \in \mathcal{B}(\mathcal{G}, \mathcal{H})$ such that $\mathrm{L}\mathrm{L}^* = \mu \mathrm{Id}$ where $\mu > 0$. Then

$$\mathrm{prox}_{f \circ \mathrm{L}} = \mathrm{Id} - \mu^{-1} \mathrm{L}^* \circ (\mathrm{Id} - \mathrm{prox}_{\mu f}) \circ \mathrm{L}.$$

Proof: $\mathrm{L}\mathrm{L}^* = \mu \mathrm{Id} \Rightarrow L = \mathcal{H}$ is closed, hence $V = (\mathrm{L}^*) = (\ker \mathrm{L})^{\perp}$ is closed. The orthogonal projection onto V is
$P_V = \mathrm{L}^*(\mathrm{L}\mathrm{L}^*)^{-1}\mathrm{L} = \mu^{-1}\mathrm{L}^*\mathrm{L}$.
For every $x \in \mathcal{H}$,
$p = \mathrm{prox}_{f \circ \mathrm{L}} x \Leftrightarrow x - p \in= \partial(f \circ \mathrm{L})(p) = \mathrm{L}^* \partial f(\mathrm{L}p)$ (since $\mathrm{L} = \mathcal{H}$).
Thus, $x - p \in V$.
It can be deduced that $P_{V^\perp} p = P_{V^\perp} x = x - P_V x = x - \mu^{-1} L^* L x$.
Furthermore,
$x - p \in \mathrm{L}^* \partial(\mathrm{L}p) \Rightarrow \mathrm{L}x - \mathrm{L}p \in \mu \partial f(\mathrm{L}p) \Leftrightarrow \mathrm{L}p = \mathrm{prox}_{\mu f}(\mathrm{L}x)$.
We have thus $P_V p = \mu^{-1}\mathrm{L}^*\mathrm{L}p = \mu^{-1}\mathrm{L}^*\mathrm{prox}_{\mu f}(\mathrm{L}x)$ and
$p = P_V p + P_{V^\perp} p = x - \mu^{-1}\mathrm{L}^*(\mathrm{Id} - \mathrm{prox}_{\mu f})(\mathrm{L}x)$.

Proximity operator: properties

Let $\mathcal{H}$ and $\mathcal{G}$ be two Hilbert spaces. Let $f \in \Gamma_0(\mathcal{H})$ and $\mathrm{L} \in \mathcal{B}(\mathcal{G}, \mathcal{H})$ such that $\mathrm{L}\mathrm{L}^* = \mu \mathrm{Id}$ where $\mu > 0$. Then

$$\mathrm{prox}_{f \circ \mathrm{L}} = \mathrm{Id} - \mu^{-1} \mathrm{L}^* \circ (\mathrm{Id} - \mathrm{prox}_{\mu f}) \circ \mathrm{L}.$$

Particular cases:

• $\mathrm{L} = \mathrm{W}$: **orthonormal wavelet transform** $\mathrm{W}\mathrm{W}^* = \mathrm{W}^*\mathrm{W} = \mathrm{Id}$, then

$$\mathrm{prox}_{f \circ \mathrm{W}} = \mathrm{W}^* \mathrm{prox}_f \mathrm{W}$$

Proximity operator: properties

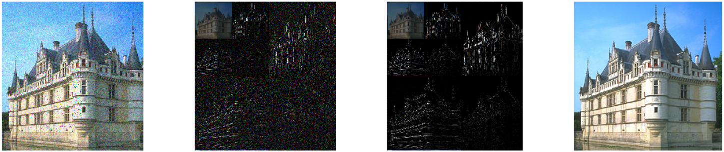
Let $\mathcal{H}$ and $\mathcal{G}$ be two Hilbert spaces. Let $f \in \Gamma_0(\mathcal{H})$ and $\mathrm{L} \in \mathcal{B}(\mathcal{G}, \mathcal{H})$ such that $\mathrm{LL}^* = \mu \mathrm{Id}$ where $\mu > 0$. Then

$$\mathrm{prox}_{f \circ \mathrm{L}} = \mathrm{Id} - \mu^{-1}\mathrm{L}^* \circ (\mathrm{Id} - \mathrm{prox}_{\mu f}) \circ \mathrm{L}.$$

Particular cases:

• $\mathrm{L} = \mathrm{W}$: **orthonormal wavelet transform** $\mathrm{WW}^* = \mathrm{W}^*\mathrm{W} = \mathrm{Id}$, then

$$\mathrm{prox}_{f \circ \mathrm{W}} = \mathrm{W}^* \mathrm{prox}_f \mathrm{W}$$



z $\qquad$ $\mathrm{u} = \mathrm{Wz}$ $\qquad$ $\mathrm{soft}_\theta(\mathrm{Wz})$ $\qquad$ $\widehat{\mathrm{x}} = \mathrm{W}^*\mathrm{soft}_\theta(\mathrm{Wz})$

Proximity operator: properties

Let $\mathcal{H}$ and $\mathcal{G}$ be two Hilbert spaces. Let $f \in \Gamma_0(\mathcal{H})$ and $\mathrm{L} \in \mathcal{B}(\mathcal{G}, \mathcal{H})$ such that $\mathrm{LL}^* = \mu \mathrm{Id}$ where $\mu > 0$. Then

$$\mathrm{prox}_{f \circ \mathrm{L}} = \mathrm{Id} - \mu^{-1} \mathrm{L}^* \circ (\mathrm{Id} - \mathrm{prox}_{\mu f}) \circ \mathrm{L}.$$

Particular cases:

- $\mathrm{L} = \mathrm{W}$: **orthonormal wavelet transform** $\mathrm{WW}^* = \mathrm{W}^*\mathrm{W} = \mathrm{Id}$, then

$$\mathrm{prox}_{f \circ \mathrm{W}} = \mathrm{W}^* \mathrm{prox}_f \mathrm{W}$$

- $\mathrm{L} = \mathrm{F}$: **tight frame** $\mathrm{F}^*\mathrm{F} = \mu \mathrm{Id}$, then

$$\mathrm{prox}_{f \circ \mathrm{F}^*} = \mathrm{Id} - \mu^{-1} \mathrm{F} \circ (\mathrm{Id} - \mathrm{prox}_{\mu f}) \circ \mathrm{F}^*$$

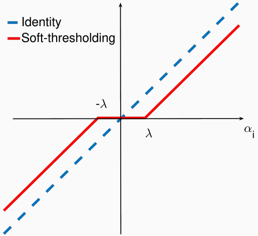
Proximal algorithm to solve $\widehat{\mathbf{x}} \in \underset{\mathbf{x}}{\operatorname{Argmin}}\ f(\mathbf{x})$

Key tool – *Proximity operator*

Backward subgradient descent: $\quad \mathrm{x}^{[k+1]} = \mathrm{x}^{[k]} - \mathrm{u}^{[k]}$ where $\mathrm{u}^{[k]} \in \tau\partial f(\mathrm{x}^{[k+1]}$

$$= \operatorname{prox}_{\tau f}(\mathrm{x}^{[k]})$$

➡ Proximity operator of the ℓ_1 norm = soft-thresholding



Proximal algorithm to solve $\widehat{\mathrm{x}} \in \underset{\mathbf{x}}{\mathrm{Argmin}}\ f(\mathbf{x})$

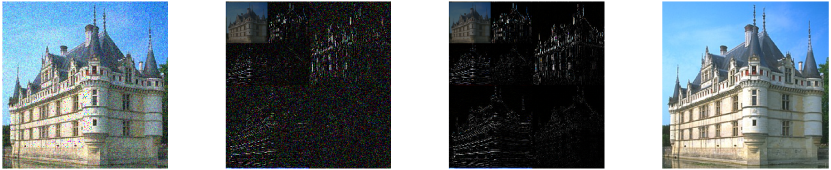
Key tool – *Proximity operator*

Backward subgradient descent: $\mathrm{x}^{[k+1]} = \mathrm{x}^{[k]} - \mathrm{u}^{[k]}$ where $\mathrm{u}^{[k]} \in \tau\partial f(\mathrm{x}^{[k+1]}$

$$= \mathrm{prox}_{\tau f}(\mathrm{x}^{[k]})$$

➡ Proximity operator of the ℓ_1 norm $=$ soft-thresholding

➡ Denoising (L orthon. basis): $\widehat{\mathrm{x}} = \mathrm{prox}_{\tau\theta\|\mathrm{L}\cdot\|_1}(\mathrm{z}) = \mathrm{L}^*\mathrm{soft}_{\gamma\theta}(\mathrm{Lz})$



z | $\mathrm{u} = \mathrm{Lz}$ | $\mathrm{soft}_{\tau\theta}(\mathrm{Lz})$ | $\widehat{\mathrm{x}} = \mathrm{L}^*\mathrm{soft}_{\theta}(\mathrm{Lz})$

Proximal algorithm to solve $\widehat{\mathrm{x}} \in \underset{\mathbf{x}}{\mathrm{Argmin}}\ f(\mathbf{x})$

Key tool – *Proximity operator*

Backward subgradient descent: $$\begin{aligned} \mathrm{x}^{[k+1]} &= \mathrm{x}^{[k]} - \mathrm{u}^{[k]} \ \text{ where } \ \mathrm{u}^{[k]} \in \tau\partial f(\mathrm{x}^{[k+1]} \\ &= \mathrm{prox}_{\tau f}(\mathrm{x}^{[k]}) \end{aligned}$$

- ➡ Proximity operator of the ℓ_1 norm $=$ soft-thresholding
- ➡ Denoising (L orthon. basis): $\widehat{\mathrm{x}} = \mathrm{prox}_{\tau\theta\|\mathrm{L}\cdot\|_1}(\mathrm{z}) = \mathrm{L}^*\mathrm{soft}_{\gamma\theta}(\mathrm{Lz})$
- ➡ Numerous closed forms: **http://proximity-operator.net**.
- ➡ ... but also **more complex operations**: $\mathrm{prox}_{f_1+f_2}$, $\mathrm{prox}_{f\circ L}$.

Proximal algorithm to solve $\widehat{\mathrm{x}} \in \underset{\mathbf{x}}{\mathrm{Argmin}}\ f(\mathbf{x})$

Key tool – *Proximity operator*

Backward subgradient descent: $\mathrm{x}^{[k+1]} = \mathrm{x}^{[k]} - \mathrm{u}^{[k]}$ where $\mathrm{u}^{[k]} \in \tau\partial f(\mathrm{x}^{[k+1]}$

$= \mathrm{prox}_{\tau f}(\mathrm{x}^{[k]})$

- Proximity operator of the ℓ_1 norm = soft-thresholding
- Denoising (L orthon. basis): $\widehat{\mathrm{x}} = \mathrm{prox}_{\tau\theta\|\mathrm{L}\cdot\|_1}(\mathrm{z}) = \mathrm{L}^*\mathrm{soft}_{\gamma\theta}(\mathrm{Lz})$
- Numerous closed forms: **http://proximity-operator.net**.
- ... but also **more complex operations**: $\mathrm{prox}_{f_1+f_2}$, $\mathrm{prox}_{f\circ L}$.

Forward-backward algorithm – $\widehat{\mathrm{x}} \in \underset{\mathrm{x}}{\mathrm{Argmin}}\ f_1(\mathrm{x}) + f_2(\mathrm{x})$

$$\mathrm{x}^{[k+1]} = \mathrm{prox}_{\tau f_2}\left(\mathrm{x}^{[k]} - \tau\nabla f_1(\mathrm{x}^{[k]})\right)$$

Proximal algorithms

➡ **Minimisation problem** :

$$\boxed{\widehat{\mathrm{x}} \in \underset{\mathrm{x}}{\mathrm{Argmin}}\ f_1(\mathrm{x}) + f_2(\mathrm{x})}$$

with f_1 and f_2 either diff. with Lipschitz gradient or proximable.

➡ **Design of a recursive sequence of the form:**

$$(\forall k \in \mathbb{N}) \qquad \mathrm{x}^{[k+1]} = \mathsf{T}\mathrm{x}^{[k]},$$

Gradient descent	$\mathsf{T} = \mathrm{Id} - \tau(\nabla f_1 + \nabla f_2)$
Proximal point	$\mathsf{T} = \mathrm{prox}_{\tau(f_1+f_2)}$
Forward-Backward	$\mathsf{T} = \mathrm{prox}_{\tau f_2}(\mathrm{Id} - \tau\nabla f_1)$
Peaceman-Rachford	$\mathsf{T} = (2\mathrm{prox}_{\tau f_2} - \mathrm{Id}) \circ (2\mathrm{prox}_{\tau f_1} - \mathrm{Id})$
Douglas-Rachford	$\mathsf{T} = \mathrm{prox}_{\tau f_2}(2\mathrm{prox}_{\tau f_1} - \mathrm{Id}) + \mathrm{Id} - \mathrm{prox}_{\tau f_1}$

Proximal algorithms

General objective function

$$\text{Find} \qquad \widehat{\mathrm{x}} \in \underset{\mathrm{x}\in\mathcal{H}}{\mathrm{Argmin}} \sum_{j=1}^{J} f_j(\mathrm{H}_j \mathrm{x})$$

where H_j denotes a linear operator from $\mathcal{H}$ to $\mathcal{G}_j$ and $(f_j)_{1\leq j\leq J}$ belong to the class of convex, l.s.c., and proper from $\mathcal{G}_j$ to $]-\infty,+\infty]$.

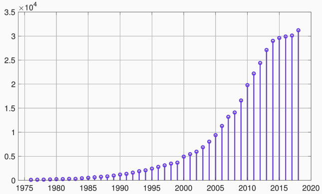
Proximal algorithms

General objective function

$$\text{Find} \qquad \widehat{\mathrm{x}} \in \underset{\mathrm{x}\in\mathcal{H}}{\mathrm{Argmin}} \sum_{j=1}^{J} f_j(\mathrm{H}_j\mathrm{x})$$

where H_j denotes a linear operator from $\mathcal{H}$ to $\mathcal{G}_j$ and $(f_j)_{1\leq j\leq J}$ belong to the class of convex, l.s.c., and proper from $\mathcal{G}_j$ to $]-\infty,+\infty]$.

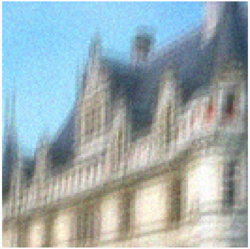
- Numerous proximal algorithms
  - Forward-Backward
  - Douglas-Rachford
  - ADMM
  - Primal-dual . . .



Number of articles per year on Google scholar containing "maximum algorithms" since 1997.

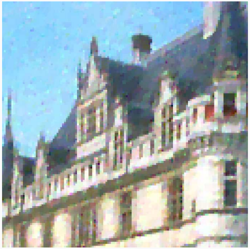
Evolution of image restoration results: blur + Gaussian noise

Original

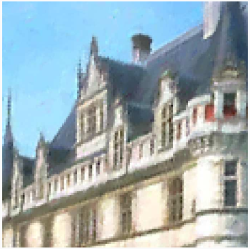


Degraded (13.4 dB)

DTT (16.6 dB)
Wavelets



TV (19.0 dB)
Finite diff



NLTV (19.4 dB)
Improved finite diff

Summary

Data: z

$\boxed{\text{Forward model}}$

$\boxed{\text{Cost function design}}$

$\boxed{\text{Minimisation algorithm}}$

$\boxed{\text{Parameter(s) selection}}$

Estimation: $\widehat{\mathrm{x}}(\mathrm{z};\widehat{\theta})$

$$\mathrm{z} = \mathcal{D}(\mathrm{A}\overline{\mathrm{x}})$$

$$\widehat{\mathrm{x}}(\mathrm{z};\theta) \in \underset{\mathrm{x}}{\mathrm{Argmin}}\ f_1(\mathrm{x};\mathrm{z}) + \theta f_2(\mathrm{x})$$

$$\mathrm{x}^{[k+1]} = \mathsf{T}\mathrm{x}^{[k]}$$

$$\widehat{\theta} \in \underset{\theta}{\mathrm{Argmin}} \|\overline{\mathrm{x}} - \widehat{\mathrm{x}}(\mathrm{z};\theta)\|_2^2$$

Towards deep learning

Data: z

Forward model

Cost function design

Minimisation algorithm

Parameter(s) selection

Estimation: $\widehat{\mathrm{x}}(\mathrm{z};\widehat{\theta})$

Data: z

Deep learning

Estimation : $\widehat{\mathrm{x}}(\mathrm{z};\widehat{\theta})$

Inversion model $\widehat{\mathrm{x}} = \mathrm{d}_\Theta(\mathrm{z})$

➡ [1922] **Maximum likelihood** (Fisher).

$$\widehat{\mathrm{x}} \in \underset{\mathrm{x}}{\mathrm{Argmin}}\ \frac{1}{2}\|\mathrm{Ax}-\mathrm{z}\|_2^2 = (\mathrm{A}^*\mathrm{A})^{-1}\mathrm{A}^*\mathrm{z}$$

➡ [1963] **Regularisation** (Tikhonov, Huber)

$$\widehat{\mathrm{x}} \in \underset{\mathrm{x}}{\mathrm{Argmin}}\ \frac{1}{2}\|\mathrm{Ax}-\mathrm{z}\|_2^2 + \theta\|\mathrm{Lx}\|_2^2 \qquad \text{avec} \quad \theta > 0$$

➡ [2000] **Sparsity** (Donoho, Daubechies-Defrise-DeMol,...)

$$\widehat{\mathrm{x}} \in \underset{\mathrm{x}}{\mathrm{Argmin}}\ \frac{1}{2}\|\mathrm{Ax}-\mathrm{z}\|_2^2 + \theta\|\mathrm{Lx}\|_\star$$

➡ [2010] **“End to end” neural networks**

$$\widehat{\mathrm{x}} = \mathrm{NN}_\Theta(\mathrm{z})$$

➡ [2020] **Model-based neural network: PnP, unrolled, . . .**

$$0 \in \mathrm{A}^*(\mathrm{A}\widehat{\mathrm{x}} - \mathrm{z}) + \mathbf{B}(\widehat{\mathrm{x}})$$

Inversion model $\widehat{x} = d_\Theta(z)$

Deep learning – *General framework*

- Dataset : $\mathcal{S} = \left\{ (\overline{x}_\ell, z_\ell) \in \mathbb{R}^N \times \mathbb{R}^M \;\middle|\; \ell \in \mathbb{I} \cup \mathbb{J} \right\}$

$\overline{x}_\ell$

z_ℓ

Inversion model $\widehat{\mathrm{x}} = \mathrm{d}_\Theta(\mathrm{z})$

Deep learning – *General framework*

- Dataset : $\mathcal{S} = \left\{ (\overline{\mathrm{x}}_\ell, \mathrm{z}_\ell) \in \mathbb{R}^N \times \mathbb{R}^M \;\middle|\; \ell \in \mathbb{I} \cup \mathbb{J} \right\}$
    - *training set* $(\overline{\mathrm{x}}_\ell, \mathrm{z}_\ell)_{\ell \in \mathbb{I}}$ with size $\mathbb{I}$
    - *testing set* $(\overline{\mathrm{x}}_\ell, \mathrm{z}_\ell)_{\ell \in \mathbb{J}}$ with size $\mathbb{J}$

Inversion model $\widehat{x} = d_\Theta(z)$

Deep learning – *General framework*

- Dataset : $\mathcal{S} = \left\{(\overline{x}_\ell, z_\ell) \in \mathbb{R}^N \times \mathbb{R}^M \;\middle|\; \ell \in \mathbb{I} \cup \mathbb{J}\right\}$
    - *training set* $(\overline{x}_\ell, z_\ell)_{\ell\in\mathbb{I}}$ with size $\mathbb{I}$
    - *testing set* $(\overline{x}_\ell, z_\ell)_{\ell\in\mathbb{J}}$ with size $\mathbb{J}$

- Prediction function: $d_\Theta(z_\ell) = \eta^{[K]}\big(W^{[K]} \dots \eta^{[1]}(W^{[1]} z_\ell + b^{[1]}) \dots + b^{[K]}\big)$
    - Linear operators: $W^{[1]}, W^{[2]}, \dots, W^{[K]}$
    - Activation functions: $\eta^{[1]}, \eta^{[2]}, \dots, \eta^{[K]}$
    - Bias: $b^{[1]}, b^{[2]}, \dots, b^{[K]}$

$\Rightarrow$ $\boldsymbol{\Theta} = \{W^{[1]}, \dots, W^{[K]}, b^{[1]}, \dots, b^{[K]}\}$

Inversion model $\widehat{x} = d_\Theta(z)$

Deep learning – *General framework*

- Dataset : $\mathcal{S} = \big\{ (\overline{x}_\ell, z_\ell) \in \mathbb{R}^N \times \mathbb{R}^M \;\big|\; \ell \in \mathbb{I} \cup \mathbb{J} \big\}$
    - *training set* $(\overline{x}_\ell, z_\ell)_{\ell \in \mathbb{I}}$ with size $\mathbb{I}$
    - *testing set* $(\overline{x}_\ell, z_\ell)_{\ell \in \mathbb{J}}$ with size $\mathbb{J}$

- Prediction function: $d_\Theta(z_\ell) = \eta^{[K]}\big(W^{[K]} \ldots \eta^{[1]}(W^{[1]} z_\ell + b^{[1]}) \ldots + b^{[K]}\big)$

$$\Rightarrow \boldsymbol{\Theta} = \{W^{[1]}, \ldots, W^{[K]}, b^{[1]}, \ldots, b^{[K]}\}$$

- Learn parameters: $\widehat{\boldsymbol{\Theta}} \in \operatorname{Argmin}_{\boldsymbol{\Theta}} \frac{1}{\mathbb{I}} \sum_{\ell \in \mathbb{I}} \mathcal{L}\Big(\overline{x}_\ell, d_{\boldsymbol{\Theta}}(z_\ell)\Big)$

Inversion model $\widehat{\mathrm{x}} = \mathrm{d}_\Theta(\mathrm{z})$

Deep learning – *General framework*

- Dataset : $\mathcal{S} = \big\{(\overline{\mathrm{x}}_\ell, \mathrm{z}_\ell) \in \mathbb{R}^N \times \mathbb{R}^M \;\big|\; \ell \in \mathbb{I} \cup \mathbb{J}\big\}$
  - *training set* $(\overline{\mathrm{x}}_\ell, \mathrm{z}_\ell)_{\ell \in \mathbb{I}}$ with size $\mathbb{I}$
  - *testing set* $(\overline{\mathrm{x}}_\ell, \mathrm{z}_\ell)_{\ell \in \mathbb{J}}$ with size $\mathbb{J}$

- Prediction function: $\mathrm{d}_\Theta(\mathrm{z}_\ell) = \eta^{[K]}(W^{[K]} \ldots \eta^{[1]}(W^{[1]}\mathrm{z}_\ell + \mathrm{b}^{[1]}) \ldots + \mathrm{b}^{[K]})$

  $\Rightarrow \boldsymbol{\Theta} = \{\mathrm{W}^{[1]}, \ldots, \mathrm{W}^{[K]}, \mathrm{b}^{[1]}, \ldots, \mathrm{b}^{[K]}\}$

- Learn parameters: $\widehat{\boldsymbol{\Theta}} \in \mathrm{Argmin}_{\boldsymbol{\Theta}} \dfrac{1}{\mathbb{I}} \sum_{\ell \in \mathbb{I}} \mathcal{L}\Big(\overline{\mathrm{x}}_\ell, \mathrm{d}_{\boldsymbol{\Theta}}(\mathrm{z}_\ell)\Big)$

- Evaluate: A properly trained network must satisfy

$$(\forall \ell \in \mathbb{J}) \quad \overline{\mathrm{x}}_\ell \approx \mathrm{d}_{\widehat{\boldsymbol{\Theta}}}(\mathrm{z}_\ell)$$

Inversion model $\widehat{\mathrm{x}} = \mathrm{d}_\Theta(\mathrm{z})$: Unrolled

Deep learning – *general framework*

- Dataset : $\mathcal{S} = \big\{(\overline{\mathrm{x}}_\ell, \mathrm{z}_\ell) \in \mathbb{R}^N \times \mathbb{R}^M \;\big|\; \ell \in \{1, \ldots, \mathbb{I}\}\big\}$
- Prediction function: $\mathrm{d}_\Theta(\mathrm{z}_\ell) = \eta^{[K]}\big(\mathrm{W}^{[K]} \ldots \eta^{[1]}(\mathrm{W}^{[1]}\mathrm{z}_\ell + b^{[1]}) \ldots + \mathrm{b}^{[K]}\big)$

Unrolled scheme – *Building an informed neural network*

- **Analysis** variational formulation: $\min_{\mathrm{x}} \frac{1}{2}\|\mathrm{A}\mathrm{x} - \mathrm{z}\|_2^2 + \theta\|\mathrm{L}\mathrm{x}\|_\star$

Inversion model $\widehat{\mathrm{x}} = \mathrm{d}_\Theta(\mathrm{z})$: Unrolled

Deep learning – *general framework*

- Dataset : $\mathcal{S} = \big\{(\overline{\mathrm{x}}_\ell, \mathrm{z}_\ell) \in \mathbb{R}^N \times \mathbb{R}^M \;\big|\; \ell \in \{1, \ldots, \mathbb{I}\}\big\}$
- Prediction function: $\mathrm{d}_\Theta(\mathrm{z}_\ell) = \eta^{[K]}(\mathrm{W}^{[K]} \ldots \eta^{[1]}(\mathrm{W}^{[1]}\mathrm{z}_\ell + b^{[1]}) \ldots + \mathrm{b}^{[K]})$

Unrolled scheme – *Building an informed neural network*

- **Analysis** variational formulation: $\min\limits_{\mathrm{x}} \frac{1}{2}\|\mathrm{Ax} - \mathrm{z}\|_2^2 + \theta\|\mathrm{Lx}\|_\star$
- Forward-backward algorithm:

$$\mathrm{x}^{[k+1]} = \mathrm{prox}_{\tau\theta\|\mathrm{L}\cdot\|_\star}\Big(\mathrm{x}^{[k]} - \tau \mathrm{A}^*(\mathrm{Ax}^{[k]} - \mathrm{z})\Big)$$

Inversion model $\widehat{x} = d_\Theta(z)$: Unrolled

Deep learning – *general framework*

- Dataset : $\mathcal{S} = \{(\overline{x}_\ell, z_\ell) \in \mathbb{R}^N \times \mathbb{R}^M \mid \ell \in \{1, \ldots, \mathbb{I}\}\}$
- Prediction function: $d_\Theta(z_\ell) = \eta^{[K]}(W^{[K]} \ldots \eta^{[1]}(W^{[1]} z_\ell + b^{[1]}) \ldots + b^{[K]})$

Unrolled scheme – *Building an informed neural network*

- **Analysis** variational formulation: $\min_x \frac{1}{2}\|Ax - z\|_2^2 + \theta\|Lx\|_\star$
- Forward-backward algorithm:

$$x^{[k+1]} = \mathrm{prox}_{\tau\theta\|L\cdot\|_\star}\left(x^{[k]} - \tau A^*(Ax^{[k]} - z)\right)$$

- **Unrolled (proximal) Neural Network:**

$$x^{[k+1]} = \underbrace{\boxed{\mathrm{prox}_{\tau\theta\|L\cdot\|_\star}}}_{\eta^{[k]}}\left(\underbrace{\boxed{\mathrm{Id} - \tau A^*A}}_{W^{[k]}}\; x^{[k]} + \underbrace{\boxed{\tau A^* z}}_{b^{[k]}}\right)$$

Inversion model $\widehat{x} = d_\Theta(z)$: Unrolled

Deep learning – *general framework*

- Dataset : $\mathcal{S} = \left\{ (\overline{x}_\ell, z_\ell) \in \mathbb{R}^N \times \mathbb{R}^M \;\middle|\; \ell \in \{1, \ldots, \mathbb{I}\} \right\}$
- Prediction function: $d_\Theta(z_\ell) = \eta^{[K]}\big(W^{[K]} \ldots \eta^{[1]}(W^{[1]} z_\ell + b^{[1]}) \ldots + b^{[K]}\big)$

Unrolled scheme – *Building an informed neural network*

- **Analysis** variational formulation: $\min_{x} \frac{1}{2}\|Ax - z\|_2^2 + \theta\|Lx\|_\star$

Inversion model $\widehat{\mathrm{x}} = \mathrm{d}_\Theta(\mathrm{z})$: Unrolled

Deep learning – *general framework*

- Dataset : $\mathcal{S} = \big\{ (\overline{\mathrm{x}}_\ell, \mathrm{z}_\ell) \in \mathbb{R}^N \times \mathbb{R}^M \;\big|\; \ell \in \{1, \ldots, \mathbb{I}\} \big\}$
- Prediction function: $\mathrm{d}_\Theta(\mathrm{z}_\ell) = \eta^{[K]}(\mathrm{W}^{[K]} \ldots \eta^{[1]}(\mathrm{W}^{[1]}\mathrm{z}_\ell + b^{[1]}) \ldots + \mathrm{b}^{[K]})$

Unrolled scheme – *Building an informed neural network*

- **Synthesis** variational formulation: $\min_{\mathrm{u}} \frac{1}{2}\|\mathrm{A}\mathrm{L}^*\mathrm{u} - \mathrm{z}\|_2^2 + \theta\|\mathrm{u}\|_\star$
- Forward-backward algorithm:

$$\mathrm{u}^{[k+1]} = \mathrm{prox}_{\tau\theta\|\cdot\|_\star}\Big(\mathrm{u}^{[k]} - \tau \mathrm{L}\mathrm{A}^*(\mathrm{A}\mathrm{L}^*\mathrm{u}^{[k]} - \mathrm{z})\Big)$$

Inversion model $\widehat{x} = d_\Theta(z)$: Unrolled

Deep learning – *general framework*

- Dataset : $\mathcal{S} = \left\{(\overline{x}_\ell, z_\ell) \in \mathbb{R}^N \times \mathbb{R}^M \;\middle|\; \ell \in \{1, \ldots, \mathbb{I}\}\right\}$
- Prediction function: $d_\Theta(z_\ell) = \eta^{[K]}\left(W^{[K]} \ldots \eta^{[1]}(W^{[1]} z_\ell + b^{[1]}) \ldots + b^{[K]}\right)$

Unrolled scheme – *Building an informed neural network*

- **Synthesis** variational formulation: $\min_{u} \frac{1}{2}\|AL^* u - z\|_2^2 + \theta \|u\|_\star$
- Forward-backward algorithm:

$$u^{[k+1]} = \operatorname{prox}_{\tau\theta\|\cdot\|_\star}\left(u^{[k]} - \tau L A^* (A L^* u^{[k]} - z)\right)$$

- **Unrolled (proximal) Neural Network:**

$$u^{[k+1]} = \underbrace{\boxed{\operatorname{prox}_{\tau\theta\|\cdot\|_\star}}}_{\eta^{[k]}}\left(\underbrace{\boxed{\mathrm{Id} - \tau L A^* A L^*}}_{W^{[k]}}\, u^{[k]} + \underbrace{\boxed{\tau L A^* z}}_{b^{[k]}}\right)$$

Inversion model $\widehat{\mathrm{x}} = \mathrm{d}_\Theta(\mathrm{z})$: Plug-and-Play (PnP)

Deep learning – *General framework*

- Dataset : $\mathcal{S} = \left\{ (\overline{\mathrm{x}}_i, \mathrm{z}_i) \in \mathbb{R}^N \times \mathbb{R}^M \;\middle|\; i \in \{1, \ldots, \mathbb{I}\} \right\}$
- Prediction function : $\mathrm{NN}_\Theta(\mathrm{z}_i) = \eta^{[K]}\big(W^{[K]} \ldots \eta^{[1]}(W^{[1]}\mathrm{z}_i + b^{[1]}) \ldots + b^{[K]}\big)$

Variational versus plug-and-play approach

- Analysis variational approach : $\min\limits_{\mathrm{x}} \frac{1}{2}\|\mathrm{Ax} - \mathrm{z}\|_2^2 + \theta\|\mathrm{Lx}\|_\star$
- Forward-backward algorithm:

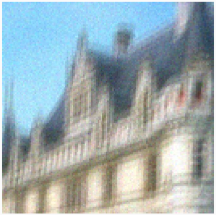
$$\mathrm{x}^{[k+1]} = \mathrm{prox}_{\tau\theta\|\mathrm{L}\cdot\|_\star}\left(\mathrm{x}^{[k]} - \tau \mathrm{A}^*(\mathrm{Ax}^{[k]} - \mathrm{z}) \right)$$

- **Plug-and-Play algorithm**:

$$\mathrm{x}^{[k+1]} = \mathrm{NN}_\Theta\left(\mathrm{x}^{[k]} - \tau \mathrm{A}^*(\mathrm{Ax}^{[k]} - \mathrm{z}) \right)$$

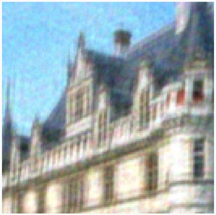
Performance summary

Original



Degraded

SNR = 13.4 dB

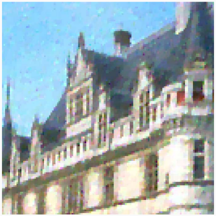


Tikhonov

SNR = 16.4 dB

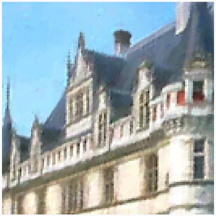
DTT

SNR = 16.6 dB



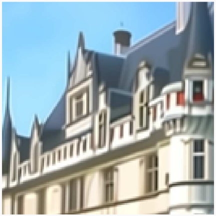
TV

SNR = 18.8 dB



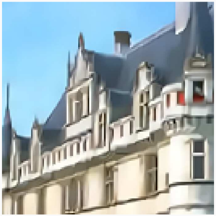
NLTV

SNR = 19.4 dB



PnP-DRUnet

SNR = 20.0 dB



PnP-ScCP

SNR = 20.2 dB

Summary of the history of image reconstruction

(Focus on reconstruction methods)