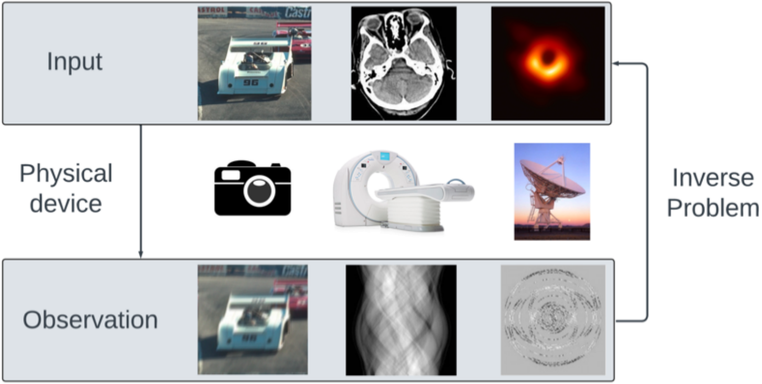
Cours 8

Image restoration in large dimension

Elisa Riccietti and Theo Mary

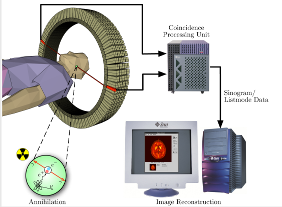
LIP-ENS Lyon

Inverse problems in imaging

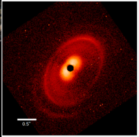


Inverse problems in imaging: various applications

Medical imaging



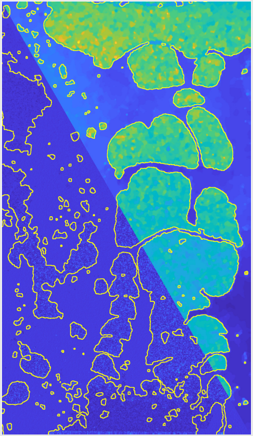
Astronomy



@ L. Denneulin

SPHERE/IRDIS

Physics



@ B. Pascal

Outline

Model the problem
- Data fidelity
- Regularization
- Quality metrics

Optimization

Applications

Neural networks in imaging

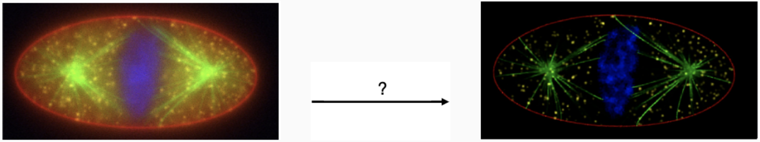
Direct model

OBSERVATION MODEL

$$z = A\bar{x} + \epsilon$$

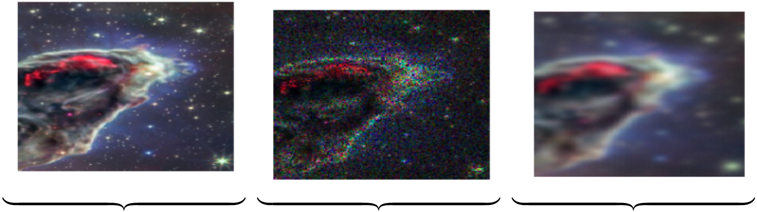
- Measurements: z
- Measurement operator: A
- Original signal: $\bar{x}$
- Additive noise: ϵ

OBJECTIVE: Find an estimate $\hat{x} \in \mathbb{R}^N$ of $\bar{x}$ from $z \in \mathbb{R}^M$



Degraded image z　　　　Original image $\bar{x}$

Examples of degradation



x Inpainting Deblurring

Inpainting: A diagonal binary
Delurring: A convolution with a blur kernel

Direct model: convolution

$$z = Ax \iff z = \phi \star x$$

- $\phi \star x$: convolution product with the Point Spread Function (PSF) ϕ of size $Q_1 \times Q_2$.
- A is a block-circulant matrix with circulant blocks related to ϕ: $A = F^* \Lambda F$ where
  - Λ: diagonal matrix,
  - F: represents the discrete Fourier transform, $*$ denotes here the transpose conjugate and $F^* = F^{-1}$.
- Efficient computation of $A\bar{x}$ by means of the Fourier transform of $\bar{x}$:

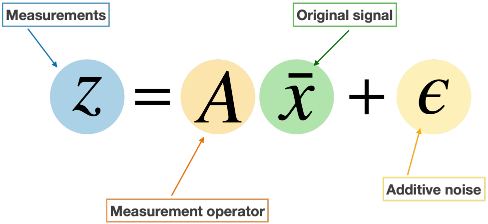
$$A\bar{x} = F^* \Lambda F \bar{x} = F^* \Lambda \bar{X}$$

Direct model: convolution

Direct model: convolution

Direct model

OBSERVATION MODEL



OBJECTIVE: Find an estimate $\hat{x} \in \mathbb{R}^N$ of $\bar{x}$ from $z \in \mathbb{R}^M$

- ☺ A is known or can be approximated
- ☺ A is sparse
- ☹ A is ill-conditioned

Hadamard conditions (1902)

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

1. **existence of a solution**: $\mathrm{range(A)} = \mathbb{R}^{\mathrm{M}}$
2. **uniqueness of the solution**: $\mathrm{Ker(A)} = 0$
3. **stability of the solution $\hat{x}$ relatively to the observation**: $\forall (z, z') \in \mathbb{R}^M$, $\|z - z'\| \to 0 \implies \|\hat{x}(z) - \hat{x}(z')\| \to 0$

Inverse problem solving

Direct inversion

$$\hat{x} = A^{-1}z = A^{-1}(A\bar{x} + \epsilon) = \bar{x} + A^{-1}\epsilon$$

Noise amplification if A is ill-conditioned

1922 **Maximum likelihood**

$$\hat{x} \in \arg\min_x \frac{1}{2}\|Ax - z\|_2^2 = (A^T A)^{-1} A^T z$$

1963 **Regularization**

$$\hat{x} \in \arg\min_x \frac{1}{2}\|Ax - z\|_2^2 + \theta\|Dx\|_2^2$$

2000 **Sparsity**

$$\hat{x} \in \arg\min_x \frac{1}{2}\|Ax - z\|_2^2 + \theta\|Dx\|_1$$

2010 **"End-to-end" neural networks**

$$\hat{x} = NN_\theta(z)$$

2020 **Model based neural networks**: PnP or Unfolded

Variational problem formulation

Estimating $\hat{x}$ close to $\bar{x}$ from the observation z and the forward operator A. Optimization formulation:

$$\widehat{x} \in \arg\min_{x} \underbrace{\frac{1}{2}\|Ax - z\|_2^2}_{\text{data fidelity}} + \underbrace{\lambda\|Lx\|_\star}_{\text{regularization}}$$

with $\|Lx\|_\star$ usually sparsity inducing norm.
The regularization permits to cope with the ill-posed nature of the inverse problem by assuming a priori knowledge about the ground-truth solution.

Examples

- L_1 norm $\|x\|_1$ to enforce sparsity
- Total Variation $\|Dx\|_1$ for promoting spacial smoothness (D differential operator)

Model the problem

Data fidelity

Regularization

Quality metrics

Optimization

Applications

Neural networks in imaging

Maximum A Posteriori (MAP) estimation

Let x and z be realizations of random variables X and Z

$$x^{MAP} = \arg\max_x \underbrace{p_{X|Z=z}(x)}_{\text{posterior distribution}}$$

By Bayes formula, the posterior distribution is given by

$$p_{X|Z=z}(x) = \frac{p_X(x)p_{Z|X=x}(z|x)}{p_Z(z)}$$

and the MAP writes as

$$x^{MAP} = \arg\min_x \underbrace{-\log p_{Z|X=x}(z)}_{\text{Data-fidelity}} \underbrace{-\log p_X(x)}_{\text{A priori}} = \arg\min_x f(x) + g(x)$$

The MAP estimator relates the data-fidelity term to the conditional distribution $p_{Z|X}$ and the regularization term to the prior distribution p_X .

Data-fidelity: Gaussian noise

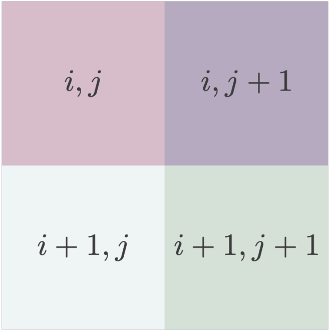
- $z = A\bar{x} + \epsilon$ with $\epsilon \sim N(0, \sigma)$
- Gaussian likelihood:

$$p_{Z|X=x}(z) = \prod_{n=1}^{M} \frac{1}{\sqrt{2\pi\sigma}} \exp(-((Ax)_n - z_n)^2/(2\sigma))$$

- L_2 data-fidelity:

$$f(x) = \sum_{n=1}^{M} \frac{1}{2\sigma}((AX)_n - z_n)^2$$

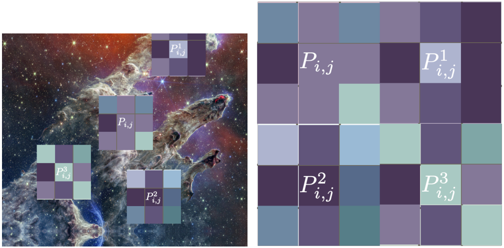
Total variation (TV)



$g(x) = g(D(x))$ contains the differences

$$\left\| \begin{bmatrix} x_{i,j} - x_{i,j+1} \\ x_{i,j} - x_{i+1,j} \end{bmatrix} \right\|$$

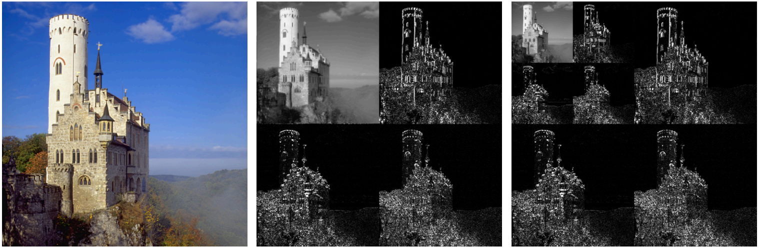
Non-local total variation (NLTV)



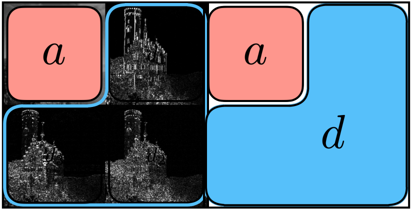
$$\left\| \begin{pmatrix} P_{i,j} - P^1_{i,j} \\ P_{i,j} - P^2_{i,j} \\ P_{i,j} - P^3_{i,j} \end{pmatrix} \right\|$$

$g(x) = g(D(x))$ contains the differences in the patches

Wavelets



Wavelets



$$x = a_0 \rightarrow \begin{cases} a_{-1} = (a_0 \star \overbrace{g}^{G}) \downarrow 2 \\ d_{-1} = (a_0 \star \overbrace{h}^{H}) \downarrow 2 \end{cases}$$

$$a_{-1} \rightarrow \begin{cases} a_{-2} = (a_{-1} \star \overbrace{g}^{G}) \downarrow 2 \\ d_{-2} = (a_{-1} \star \overbrace{h}^{H}) \downarrow 2 \end{cases}$$

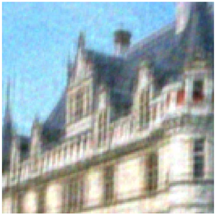
g : low pass filter

h : high pass filter

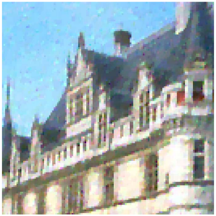
Comparison of regularizations

Original

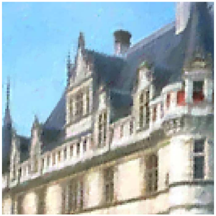
Degraded
SNR = 13.4 dB



Tikhonov
SNR = 16.4 dB



TV
SNR = 18.8 dB



NLTV
SNR = 19.4 dB

DTT
SNR = 16.6 dB

Model the problem

Data fidelity

Regularization

Quality metrics

Optimization

Applications

Neural networks in imaging

Image quality metrics

- Visual quality of the reconstruction
- Mean Squared Error (MSE) (should be as low as possible):

$$MSE(x, \hat{x}) = \frac{1}{N} \sum_{i=1}^{N} (\hat{x}_i - x_i)^2$$

- Signal-to-Noise Ratio (SNR): compares the energy of the original image to the energy of the noise (dB=decibel)

$$SNR_{dB}(x, \hat{x}) = 10 \log_{10} \frac{P_{\text{signal}}}{P_{\text{noise}}} = 10 \log_{10} \frac{\|\hat{x}\|^2}{\|\hat{x} - x\|^2}$$

  should be as high as possible.
- Peak Signal-to-Noise Ratio (PSNR):

$$PSNR(x, \hat{x}) = 10 \log_{10} \frac{N \max^2 x}{\|\hat{x} - x\|^2}$$

  ratio between the maximum intensity and the mean squared error. It can be useful when images have high dynamic range (i.e., large differences in intensity).

Outline

Problem formulation

The image restoration problem can be generally stated as

$$\min_x \; f(x) + g(x)$$

- f differentiable with Lipschitz gradient
- g possibly non-smooth but proximable

Classical solution methods:

- Proximal methods

Proximal methods

$$\min_x f(x) + \lambda g(x)$$

Assumptions:

- $f + \lambda g$ admits a minimizer
- f, g are convex
- f is L_f-smooth: $\|\nabla f(x) - \nabla f(y)\| \leq L_f \|x - y\|$ for all x, y
- g is possibly non-differentiable

Smooth case

If $g = 0$

Gradient descent: $x_{k+1} = x_k + p_k := x_k - \frac{1}{L_f} \nabla f(x_k)$

Why?

Gradient step minimizes an upper bound on the function:

$$f(x) \leq f(y) + \nabla f(y)^T (x - y) + \frac{L_f}{2} \|x - y\|^2, \quad \forall x, y$$

$$f(x_{k+1}) \leq f(x_k) + \nabla f(x_k)^T p_k + \frac{L_f}{2} \|p_k\|^2 \overbrace{\rightarrow}^{\arg\min_{p_k}} p_k = -\frac{\nabla f(x_k)}{L_f}$$

Proximal methods

If there is g: add λg to the upper bound:

$$f(x) + \lambda g(x) \leq f(y) + \nabla f(y)^T(x-y) + \frac{L_f}{2}\|x-y\|^2 + \lambda g(x), \quad \forall x, y$$

Can we minimize the upper bound?

$$\begin{aligned}
&\arg\min_x f(y) + \nabla f(y)^T(x-y) + \frac{L_f}{2}\|x-y\|^2 + \lambda g(x) = \\
&\arg\min_x \nabla f(y)^T(x-y) + \frac{L_f}{2}\|x-y\|^2 + \lambda g(x) = \\
&\arg\min_x \frac{1}{2}\|x - (y - \frac{1}{L_f}\nabla f(y))\|^2 + \frac{\lambda}{L_f} g(x) := \\
&\mathrm{prox}_{\frac{\lambda}{L_f} g}(y - \frac{1}{L_f}\nabla f(y)).
\end{aligned}$$

Proximal operator

- If g is a proper, lower semi-continuous convex function from a Hilbert space X to $[-\infty, +\infty]$ the proximal operator is defined as :

$$\mathrm{prox}_{\tau g}(y) = \arg\min_x \frac{1}{2}\|x - y\|^2 + \tau g(x)$$

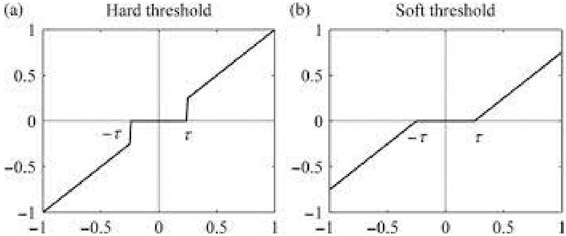
- The proximal operator can be seen as a generalization of the projection operator: if f is the characteristic function ι_C of a nonempty, closed, convex set C

$$\begin{aligned} \mathrm{prox}_{\iota_C}(x) &= \mathrm{argmin}_y \begin{cases} \frac{1}{2}\|x - y\|_2^2 & \text{if } y \in C \\ +\infty & \text{if } y \notin C \end{cases} \\ &= \mathrm{argmin}_{y \in C} \frac{1}{2}\|x - y\|_2^2 \end{aligned}$$

Example: Thresholding

$$H_t(x) = x \cdot I(|x| > t) \quad S_t(x) = sign(x)(|x| - t)_+$$

are the *Hard and soft thresholding functions*.



- If $g(x) = \|x\|_1$, $\mathrm{prox}_{\lambda g}(x) = S_\lambda(x)$
- If $h(x) = \|x\|_0$, $\mathrm{prox}_{\lambda g}(x) = H_\lambda(x)$

Gradient methods

Differentiable case

$$x_{k+1} = x_k - \frac{1}{L_f} \nabla f(x_k)$$

Convergence

If f is differentiable, L_f-smooth and convex:

$$f(x_K) - f(x^*) \leq \frac{2L_f \|x_1 - x^*\|}{K - 1}$$

Proximal gradient descent

$$x_{k+1} = \mathrm{prox}_{\lambda g/L_f}(x_k - 1/L_f \nabla f(x_k))$$

Convergence

If f is differentiable, L_f-smooth, convex and g is convex:

$$f(x_K) - f(x^*) \leq \frac{L_f \|x_1 - x^*\|}{2K}$$

In both cases $f(x_K) - f(x^*) = O\left(\frac{1}{K}\right)$.

Accelerated proximal methods: FISTA

ISTA (Iterative Shrinkage-Thresholding Algorithm):

$$x_{k+1} = \underbrace{\mathrm{prox}_{\tau g}}_{\text{step on } g} \underbrace{\left(x_k - \tau \nabla f(x_k)\right)}_{\text{step on } f}$$

Rate: $O(1/k)$ if $\tau \leq \frac{1}{L_f}$

FISTA (Fast ISTA):

$$x_{k+1} = \mathrm{prox}_{\tau g}\left(y_k - \tau \nabla f(y_k)\right)$$
$$y_{k+1} = x_k + \alpha_k (x_{k+1} - x_k)$$

where α_k is chosen as $\alpha_k = \frac{t_k - 1}{t_{k+1}}$, with $t_{k+1} = \left(\frac{k+a}{a}\right)^d$.

Rate: $O(1/k^2)$ if $\tau \leq \frac{1}{L_f}$

Limitations of proximal methods

- require prox computation (usually not available in closed form: TV, NLTV)
- suitable for problems of reasonable size: high dimensional problems $\rightarrow$ high computation time.

Main goal : provide acceleration for high dimensional problems.

Alternatives:

- FISTA [Beck & Teboulle, 2009],
- Preconditioning [Donatelli, 2019],
- Block methods [Liu, 1996],
- Exploit the problem structure with a multiresolution strategy

ML to leverage large dimensions?

Rappel: multilevel methods

ML approaches for nonlinear smooth problems

- S.G. Nash, MG/Opt (2000)
- S. Gratton, A. Sartenaer, and P. Toint, RMTR (2008)

Multilevel methods for imaging problems?

smooth ML approaches on smoothed image problems

- A. Javaherian and S. Holman, (tomography, 2017)
- S. W. Fung and Z. Wendy, (phase retrieval, 2020)
- J. Plier, F. Savarino, M. Kocvara, and S. Petra, (tomography, 2021)

Idea

$$\min_x \; f(x) + g(x) \sim \min_x f(x) + g_\gamma(x)$$

with g_γ differentiable $\implies$ not SOTA reconstruction

Multilevel methods for imaging problems?

smooth ML approaches on smoothed image problems

- A. Javaherian and S. Holman, (tomography, 2017)
- S. W. Fung and Z. Wendy, (phase retrieval, 2020)
- J. Plier, F. Savarino, M. Kocvara, and S. Petra, (tomography, 2021)

Idea

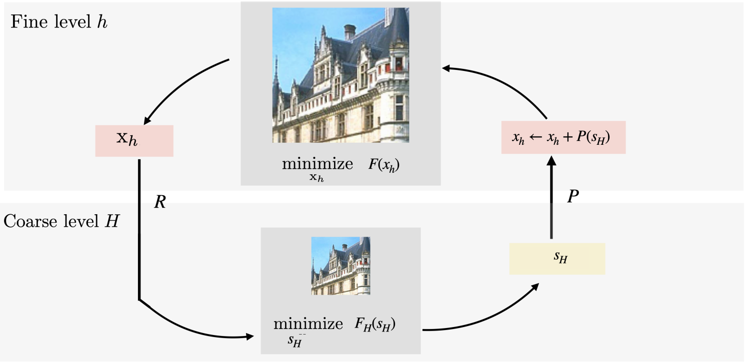
$$\min_x \; f(x) + g(x) \sim \min_x f(x) + g_\gamma(x)$$

with g_γ differentiable $\implies$ not SOTA reconstruction

Extension of ML to a non-smooth setting?

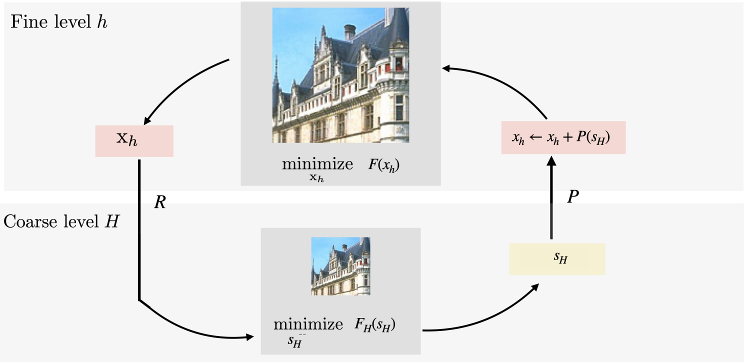
An iteration of a multilevel procedure

$$\min_{x} F(x) = f(x) + g(x)$$



An iteration of a multilevel procedure

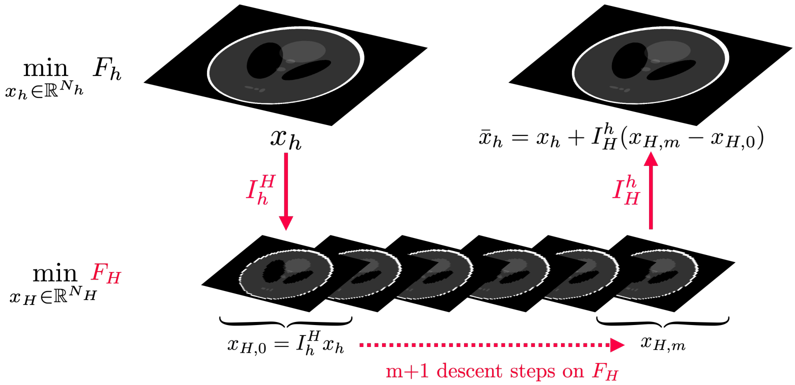
$$\min_x F(x) = f(x) + g(x)$$



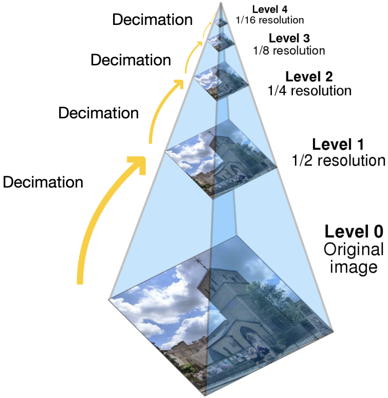
$R(= I_h^H), P = (I_H^h)$?

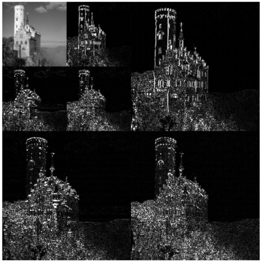
F_H?

An iteration of a multilevel procedure



A hierarchy of images





Coarse model definition F_H

$$F(x) = \frac{1}{2}\|Ax - z\|_2^2 + \lambda\|Lx\|_1$$

$$F_H(x) \overbrace{=}^{?} \frac{1}{2}\|A_H x_H - z\|_2^2 + \lambda\|L_H x_H\|_1$$

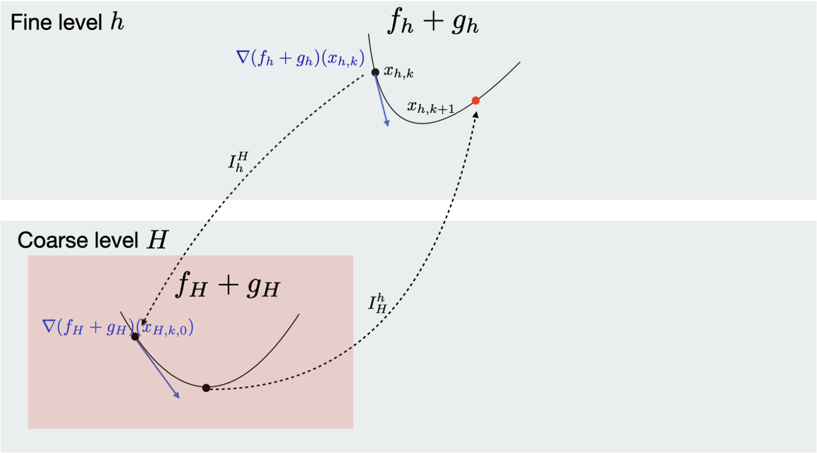
Coarse model definition F_H

$$F(x) = \frac{1}{2}\|Ax - z\|_2^2 + \lambda\|Lx\|_1$$

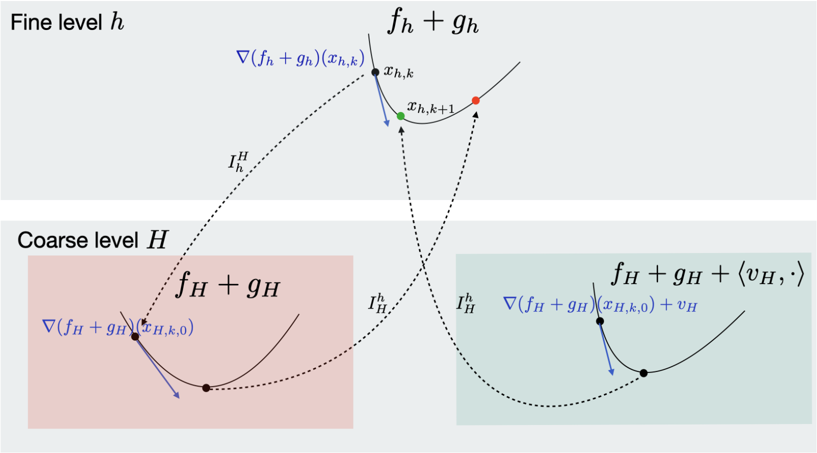
$$F_H(x) \overbrace{=}^{?} \frac{1}{2}\|A_H x_H - z\|_2^2 + \lambda\|L_H x_H\|_1$$

Is this model useful in minimizing F?

Design of F_H in smooth context: First order coherence



Design of F_H in smooth context: First order coherence



Coarse model definition F_H

If g was smooth:

$$F(x) = \frac{1}{2}\|Ax - z\|_2^2 + \lambda\|Lx\|_1$$

$$F_H(x_H) = \frac{1}{2}\|A_H x_H - z\|_2^2 + \lambda\|L_H x_H\|_1 + < v_H, x_H >$$

$$v_H = I_h^H \nabla F(x) - \nabla F_H(I_h^H x)$$

Coarse model definition F_H

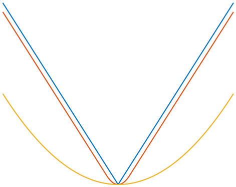
If g was smooth:

$$F(x) = \frac{1}{2}\|Ax - z\|_2^2 + \lambda\|Lx\|_1$$

$$F_H(x_H) = \frac{1}{2}\|A_H x_H - z\|_2^2 + \lambda\|L_H x_H\|_1 + < v_H, x_H >$$

$$v_H = I_h^H \nabla F(x) - \nabla F_H(I_h^H x)$$

But g is nonsmooth → smoothing! [Parpas 2017]



Smoothed convex function

Definition *[Beck 2012, Definition 2.1]*

Let g be a convex, l.s.c., and proper function on $\mathbb{R}^N$. For every $\gamma > 0$, g_γ is a smoothed convex function if there exist scalars η_1, η_2 satisfying $\eta_1 + \eta_2 > 0$ such that the following holds:

$$(\forall y \in \mathbb{R}^N) \qquad g(y) - \eta_1 \gamma \leq g_\gamma(y) \leq g(y) + \eta_2 \gamma.$$

Example: the Moreau envelope

$$g_\gamma(\cdot) = \inf_y g(y) + \frac{1}{2\gamma} \| \cdot - y \|^2$$

First-order coherence in non-smooth case

$$F_H(x_H) = f_H(x_H) + g_{H,\gamma_H}(x_H) + \langle v_H, x_H \rangle$$
$$v_H = I_h^H \nabla \left(f_h + g_{h,\gamma_h} \right)(x_h) - \nabla \left(f_H + g_{H,\gamma_H} \right)(I_h^H x_h).$$

Smooth coarse model: allows for smooth optimizers at coarse level

Fine level function decrease

Lemma

Assume that $I_H^h = \alpha (I_h^H)^T$, $\alpha > 0$. If $x_{H,m} - x_{H,0}$ is a descent direction for the coarse model and the first-order coherence holds, $I_H^h(x_{H,m} - x_{H,0})$ is a descent direction for $F_{h,\gamma_h} := f_h + g_{h,\gamma_h}$.

Proof.

???

□

What happens to non-smooth objective function?

Lemma

Under the same assumptions as in the previous lemma,

$$F_h(x_h + \bar{\tau} I_h^H (x_{H,m} - x_{H,0})) \leq F_h(x_h) + (\eta_1 + \eta_2)\gamma_h$$

for $\bar{\tau} > 0$.

Proof.

??? □

To summarize: a multilevel method for non-smooth problems

- Fine function:

$$F_h = f_h + g_h$$

  with g_h non-smooth.

- Coarse model:

$$F_H(x_H) = f_H(x_H) + g_{H,\gamma_H}(x_H) + \langle v_H, x_H \rangle$$

$$v_H = I_h^H \nabla \left(f_h + g_{h,\gamma_h} \right)(x_h) - \nabla \left(f_H + g_{H,\gamma_H} \right)(I_h^H x_h).$$

- Iterations:

$$x_{k+1} = \text{prox}_{\tau g}\left(\bar{x}_k - \tau \nabla f(\bar{x}_k) \right)$$

$$\bar{x}_k = ML(x_k) \Longleftrightarrow \min F_H(\text{ smooth})$$

IML FISTA: inertial multilevel FISTA with inexact prox

- $g(x) = \varphi(Lx)$
- Inexact proximal steps to handle state-of-the-art regularization: TV, NLTV

$$x_{k+1} \sim \mathrm{prox}_{\tau\varphi\circ \mathrm{L}}\left(\bar{y}_k - \tau\nabla f(\bar{y}_k)\right)$$
$$y_{k+1} = x_{k+1} + \alpha_k(x_{k+1} - x_k)$$

- FISTA acceleration
  - FISTA: $\bar{y}_k = y_k$
  - IML FISTA: $\bar{y}_k = ML(y_k) \iff \min F_H$
- Obtain state-of-the-art convergence guarantees (rate $O(1/k^2)$) plus dimension reduction

Multilevel algorithm for nonsmooth optimization

Set $x_{h,0}, y_{h,0} \in \mathbb{R}^N$, $t_{h,0} = 1$

while Stopping criterion is not met **do**

  if Descent condition and $r < p$ **then**

    $r = r + 1$,

    $x_{H,0} = I_h^H y_{h,k}$ Projection

    $x_{H,m} = \Phi_H \circ .. \circ \Phi_H(x_{H,0})$ Coarse minimization

    Set $\bar{\tau}_h > 0$,

    $\bar{y}_{h,k} = y_{h,k} + \bar{\tau}_h I_h^H \left(x_{H,m} - x_{H,0}\right)$ Coarse step update

  else

    $\bar{y}_{h,k} = y_{h,k}$

  end if

  $x_{h,k+1} = \Phi_i^{\epsilon_{h,k}}(\bar{y}_{h,k})$ Inexact Forward-backward step

  $t_{h,k+1} = \left(\frac{k+a}{a}\right)^d$, $\alpha_{h,k} = \frac{t_{h,k}-1}{t_{h,k+1}}$

  $y_{h,k+1} = x_{h,k+1} + \alpha_{h,k}(x_{h,k+1} - x_{h,k})$. Inertial step

end while=0

Inexact proximal step

The ϵ-subdifferential of g at $z \in \text{dom } g$ is defined as:

$$\partial_\epsilon g(z) = \{y \in \mathbb{R}^N \mid g(x) \geq g(z) + \langle x - z, y \rangle - \epsilon, \forall x \in \mathbb{R}^N\}.$$

Type 0 **approximation** [Combettes, Wajs, 2005]
$z \in \mathbb{R}^N$ is a type 0 approximation of $\text{prox}_{\gamma g}(y)$ with precision ϵ, and we write $z \simeq_0 \text{prox}_{\gamma g}(y)$, if and only if $\|z - \text{prox}_{\gamma g}(y)\| \leq \sqrt{2\gamma\epsilon}$.

Type 1 **approximation** [Villa et al., 2013]
$z \in \mathbb{R}^N$ is a type 1 approximation of $\text{prox}_{\gamma g}(y)$ ith precision ϵ, and we write $z \simeq_1 \text{prox}_{\gamma g}(y)$, if and only if $0 \in \partial_\epsilon \left(g(z) + \frac{1}{2\gamma}\|z - y\|^2 \right)$.

Type 2 **approximation** [Villa et al., 2013]
$z \in \mathbb{R}^N$ is a type 2 approximation of $\text{prox}_{\gamma g}(y)$ with precision ϵ, and we write $z \simeq_2 \text{prox}_{\gamma g}(y)$, if and only if $\frac{y-z}{\gamma} \in \partial_\epsilon g(z)$.

Example

$$\operatorname{prox}_{\tau g}(y) = \arg\min_{x} \frac{1}{2}\|x - y\|^2 + \tau g(x)$$

Inexact solution via an iterative method

Convergence analysis

Theorem 1

Considering $\forall k \in \mathbb{N}^*,\ \alpha_{h,k} = 0$ and the sequence $(\epsilon_{h,k})_{k\in\mathbb{N}}$ is such that $\sum_{k\in\mathbb{N}} \sqrt{\|\epsilon_{h,k}\|} < +\infty$. Set $x_{h,0} \in \mathbb{R}^{N_h}$ and choosing approximation of Type 0, the sequence $(x_{h,k})_{k\in\mathbb{N}}$ generated by IML FISTA converges to a minimizer of F_h.

Theorem 2

Let $\forall k \in \mathbb{N}^*,\ t_{h,k+1} = \left(\frac{k+a}{a}\right)^d$, with (a, d) satisfying the conditions in [Aujol, Dossal, 2015 – Definition 3.1]. Moreover, if we assume that:

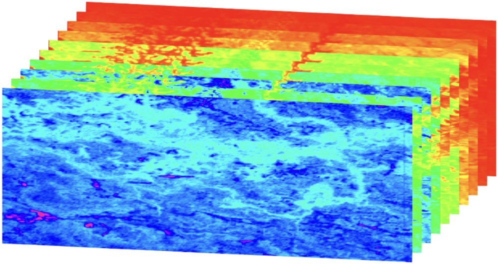
- $\sum_{k=1}^{+\infty} k^d \sqrt{\epsilon_{h,k}} < +\infty$ in the case of Type 1 approximation,
- $\sum_{k=1}^{+\infty} k^{2d} \epsilon_{h,k} < +\infty$ in the case of Type 2 approximation.

Let $(x_{h,k})_{k\in\mathbb{N}}$ the sequence generated by IML FISTA, then

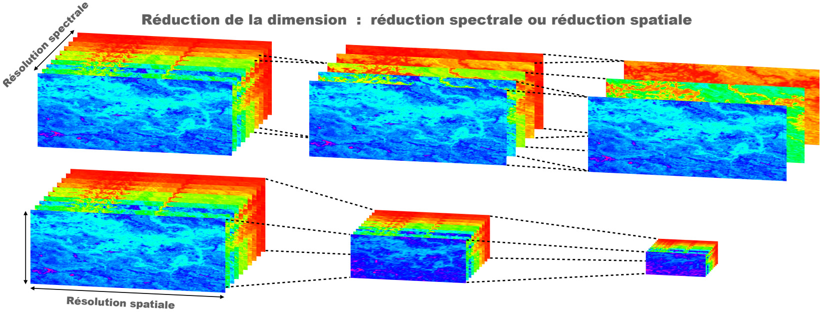
- The sequence $\left(k^{2d}\left(F_h(x_{h,k}) - F_h(x^*)\right)\right)_{k\in\mathbb{N}}$ belongs to $\ell_\infty(\mathbb{N})$.
- The sequence $(x_{h,k})_{k\in\mathbb{N}}$ converges to a minimizer of F_h.

Outline

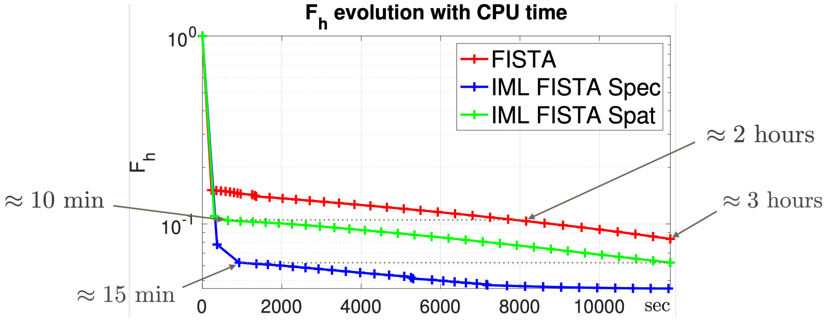
Hyperspectral images

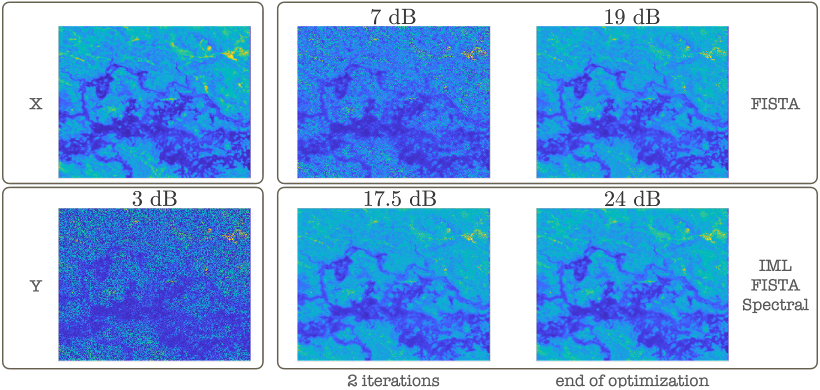


How to build the coarse approximations?

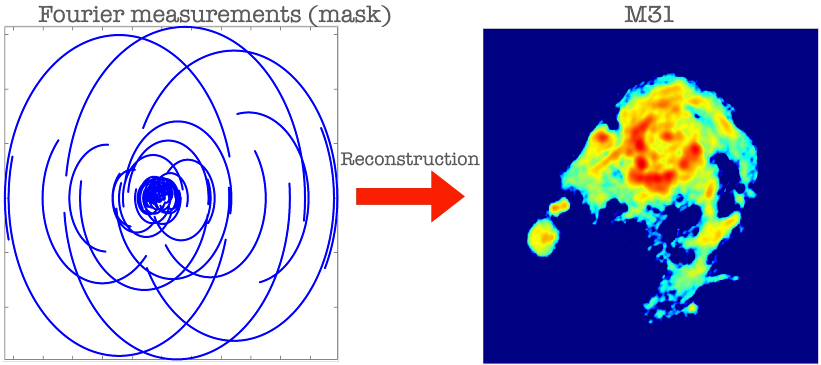


Objective function evolution



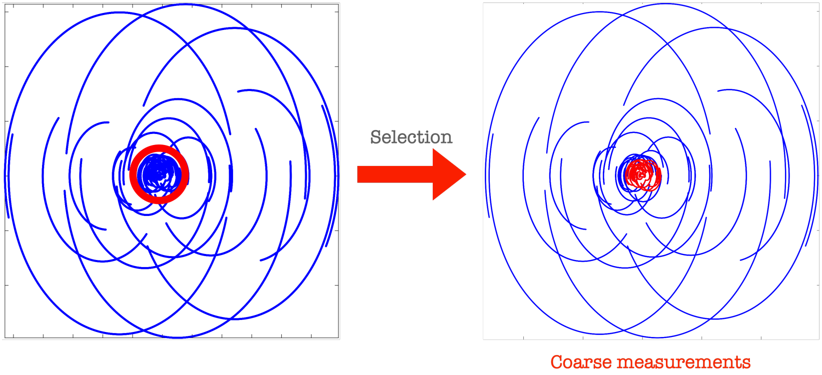

X
7 dB
19 dB
FISTA
Y
3 dB
17.5 dB
24 dB
IML
FISTA
Spectral
2 iterations
end of optimization

Radio-interferometric imaging

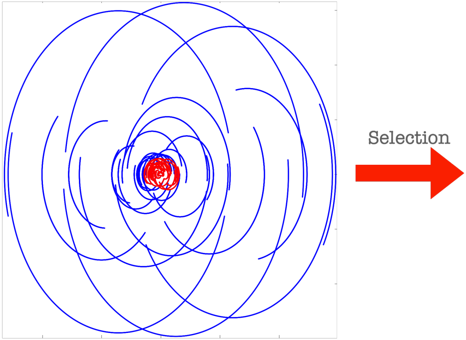


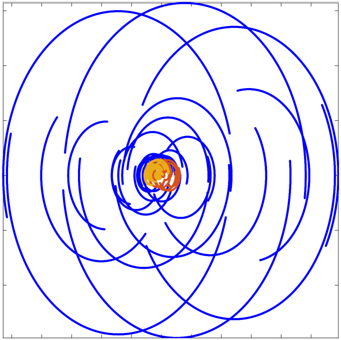
Dimension bottleneck: number of observations

Radio-interferometric imaging



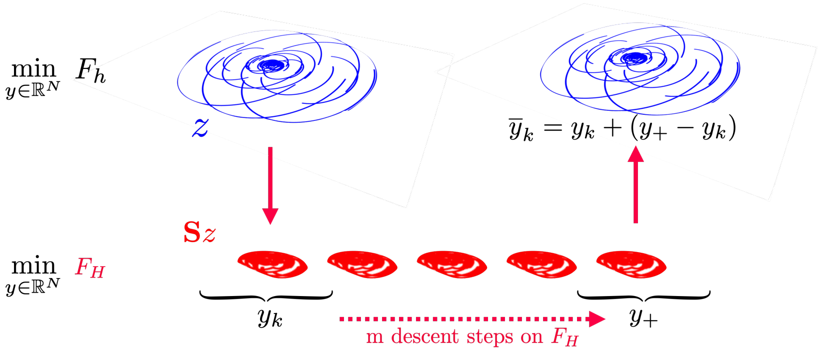
Radio-interferometric imaging



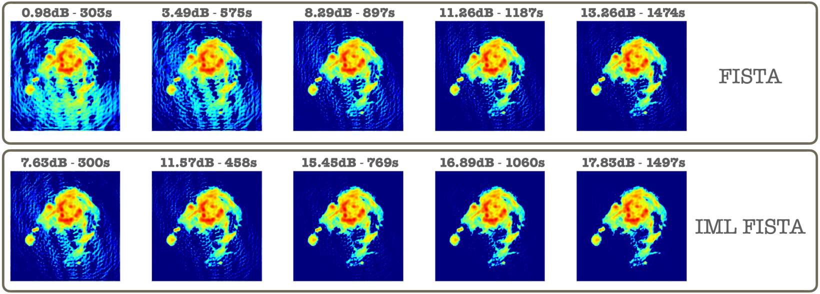


Coarser measurements

Radio-interferometric imaging

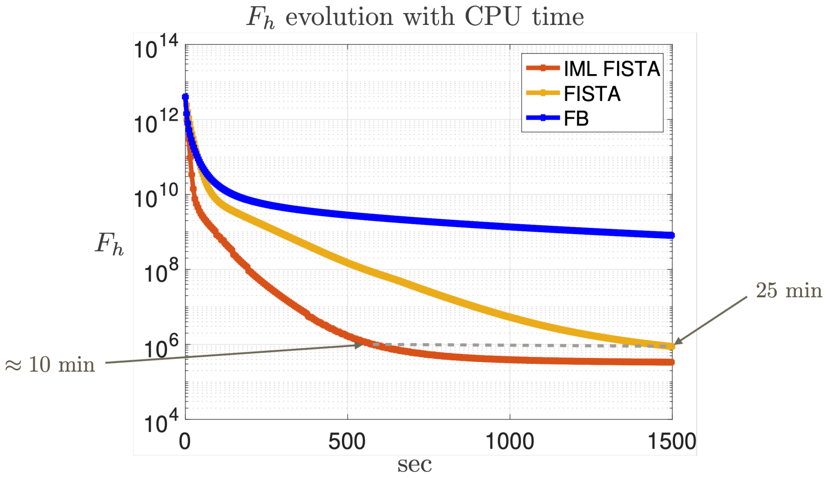


Radio-interferometric imaging



Reconstruction in log-scale of a region of the M31 galaxy

Radio-interferometric imaging



Outline

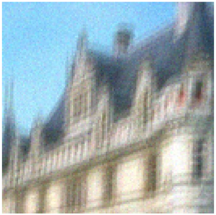
Neural networks in imaging

How to exploit the great power of neural networks in imaging?
Basically three approaches:

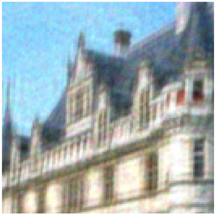
1. End-to-end
2. Plug and Play
3. Unrolled algorithms

Example

Original

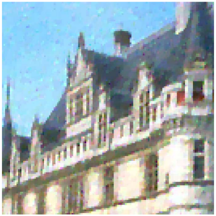


Degraded
SNR = 13.4 dB

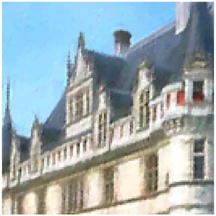


Tikhonov
SNR = 16.4 dB

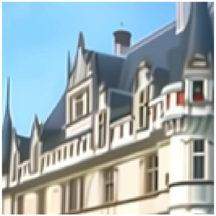
DTT
SNR = 16.6 dB



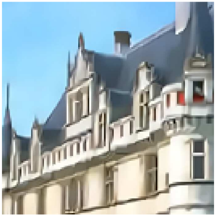
TV
SNR = 18.8 dB



NLTV
SNR = 19.4 dB



PnP-DRUnet
SNR = 20.0 dB



PnP-ScCP
SNR = 20.2 dB

1) End-to-end approaches

Problem: $z = A\bar{x} + \epsilon$

Idea: model $\hat{x}$ by a neural network

- Build a dataset $\mathcal{D} = \{(\bar{x}^i, z^i)\}$ $i = 1, \ldots, m$, from the model $z = A\bar{x} + \epsilon$
- Model $\hat{x} = F_\theta(z)$ or $\hat{x} = F_\theta((A^T A)^{-1} A^T z)$ (CNN, Unet)
- Train F_θ on $\mathcal{D}$ (loss: $\ell(\bar{x}^i, z^i)$)
- Feed F_θ a noisy image, get the denoised approximation

☺ Very good denoising performance
☹ Expensive, need new training for every model

2) Plug-and-Play (PnP)

Idea: replace proximal step by a trained denoiser

$$x_{k+1} = \mathrm{prox}_{\tau g}(x_k - \tau \nabla f(x_k)) \to x_{k+1} = D(x_k - \tau \nabla f(x_k))$$

- Build a dataset $\mathcal{D} = \{(\bar{x}^i, y^i)\}$ $i = 1, \dots, m$, from the model $y = \bar{x} + \epsilon$
- Train a denoiser D (loss: $\ell(\bar{x}^i, y^i)$)
- Perform optimization steps $x_{k+1} = D(x_k - \tau \nabla f(x_k))$

☺ No need of retraining D on your problem, works for general problems. Theoretical results
☹ Less good results than end-to-end (but better than classical methods)

Convergence of PnP

$$x_{k+1} = \mathrm{prox}_{\tau g}(x_k - \tau \nabla f(x_k)) \rightarrow x_{k+1} = D(x_k - \tau \nabla f(x_k))$$

- From monotone operator theory, if f convex, provided that $\tau < \frac{2}{L_f}$, FB is guaranteed to converge to a solution of $f + g$.
- If f non-convex, provided that $\tau < \frac{1}{L_f}$, FB is guaranteed to converge to a solution of $f + g$.
- If D is trained to be a firmly non-expansive operator (by regularizing the training loss) the algorithm converges

MMSE interpretation of PnP

By definition

$$\hat{x}_{MMSE} = \arg\min_{x} \mathbb{E}(x|Z = z)$$

If we train D_θ to minimize

$$\frac{1}{m}\sum_{i=1}^{m} \| D_\theta(y^i) - \bar{x}^i \|^2$$

we can see that

$$D_{\theta^*} \sim \hat{x}_{MMSE}$$

3) Unrolled algorithms

Background: structure of a neural network

$$F_\theta(x) = \eta_k(W_k \eta_{k-1}(W_{k-1} \ldots . \eta_1(W_1 x + b_1) ..) + b_k)$$

Most activation functions are proximity operators : $\eta_k = \mathrm{prox}\Phi$
https://proximity-operator.net/

3) Unrolled algorithms

Problem formulation:

$$\arg\min_{\alpha} \frac{1}{2}\|AD^T\alpha - z\|^2 + \lambda\|\alpha\|_1$$

Iterations:

$$\begin{aligned}\alpha_{k+1} &= \mathrm{prox}_{\tau\|\cdot\|_1}(\alpha_k - \tau DA^T(AD\alpha_k - z)) \\ &= \underbrace{\mathrm{prox}_{\tau\|\cdot\|_1}}_{\eta_k}[\underbrace{(I - \tau DA^T AD^T)}_{W_k}\alpha_k + \underbrace{\tau DA^T z}_{b_k})]\end{aligned}$$

$$F_\theta(x) = \eta_k(W_k\eta_{k-1}(W_{k-1}....\eta_1(W_1 x + b_1)..) + b_k)$$

Towards multilevel PnP

$$f(x) = \frac{1}{2}\|Ax - b\|^2$$

Fine level:

$$x_{k+1}^h = D^h(x_k^h - \tau\nabla f^h(x_k^h))$$

Coarse level:

- Reduce the image size: $x_{k+1}^H = D(x_k^H - \tau\nabla f^H(x_k^H))$

Towards multilevel PnP

$$f(x) = \frac{1}{2}\|Ax - b\|^2$$

Fine level:

$$x_{k+1}^h = D^h(x_k^h - \tau \nabla f^h(x_k^h))$$

Coarse level:

- Reduce the image size: $x_{k+1}^H = D(x_k^H - \tau \nabla f^H(x_k^H))$
- Reduce the network size: $x_{k+1}^H = D^H(x_k^H - \tau \nabla f^H(x_k^H))$

Towards multilevel PnP

$$f(x) = \frac{1}{2}\|Ax - b\|^2$$

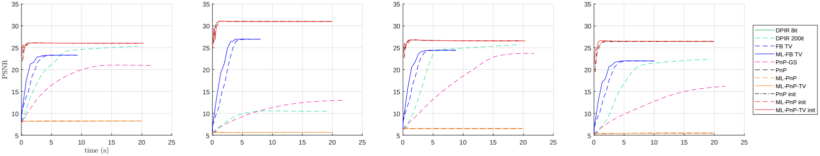
Fine level:

$$x_{k+1}^h = D^h(x_k^h - \tau \nabla f^h(x_k^h))$$

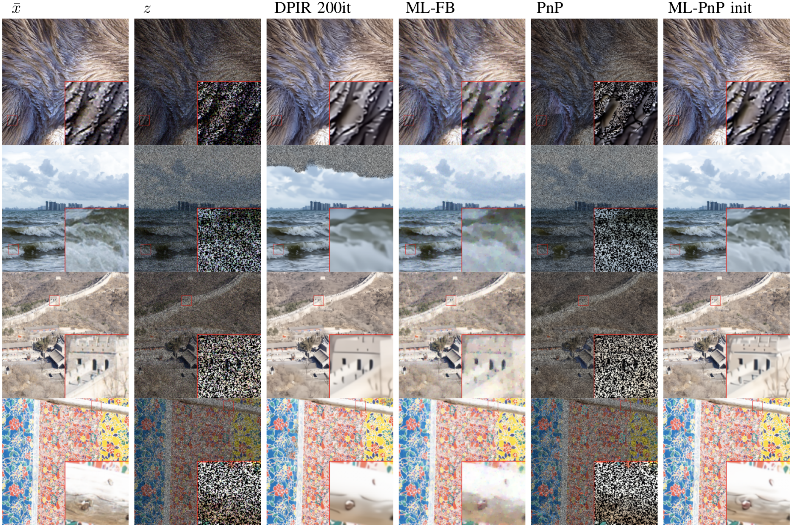
Coarse level:

- Reduce the image size: $x_{k+1}^H = D(x_k^H - \tau \nabla f^H(x_k^H))$
- Reduce the network size: $x_{k+1}^H = D^H(x_k^H - \tau \nabla f^H(x_k^H))$
- Simplify coarse iterations: $x_{k+1}^H = \mathrm{prox}_g(x_k^H - \tau \nabla f^H(x_k^H))$

Numerical tests: inpainting



Numerical tests: inpainting



Exercise

1. Complete the ??? proofs and type them in latex
2. Complete the notebook