Mathematics of Computational Imaging Systems

Unrolled/Unfolded

Nelly Pustelnik

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

Focus on the following issues

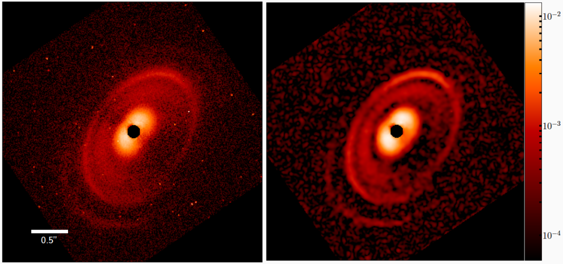
- Faster algorithm, better unfolded neural network ?
- Robustness of unfolded networks.
- Number of parameters versus performances.

Context: Image restoration

➡ **Data**: $\mathrm{z} \in \mathbb{R}^M$ degraded version of an original image $\overline{\mathrm{x}} \in \mathbb{R}^N$:

$$\boxed{\mathrm{z} = \mathrm{A}\overline{\mathrm{x}} + \varepsilon}$$

- $\mathrm{A} \in \mathbb{R}^{M \times N}$: linear degradation (e.g. a blur)
- ε: noise (e.g. Gaussian noise)



SPHERE-IRDIS

Training a prediction function for a reconstruction task

Deep learning – *General framework*

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

$\overline{\mathrm{x}}_\ell$

z_ℓ

Training a prediction function for a reconstruction task

Deep learning – *General framework*

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

Training a prediction function for a reconstruction task

Deep learning – *General framework*

- Dataset : $\mathcal{S} = \left\{ (\overline{\mathrm{x}}_\ell, \mathrm{z}_\ell) \in \mathbb{R}^N \times \mathbb{R}^M \;\middle|\; \ell \in \mathbb{I} \cup \mathbb{J} \right\}$ with $\mathrm{z}_\ell = \mathrm{A}\overline{\mathrm{x}}_\ell + \varepsilon_\ell$
    - *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: $\mathrm{d}_\Theta(\mathrm{z}_\ell) = \eta^{[K]}\big(W^{[K]} \ldots \eta^{[1]}(W^{[1]}\mathrm{z}_\ell + \mathrm{b}^{[1]}) \ldots + \mathrm{b}^{[K]}\big)$
    - Linear operators: $\mathrm{W}^{[1]}, \mathrm{W}^{[2]}, \ldots, \mathrm{W}^{[K]}$
    - Activation functions: $\eta^{[1]}, \eta^{[2]}, \ldots, \eta^{[K]}$
    - Bias: $\mathrm{b}^{[1]}, \mathrm{b}^{[2]}, \ldots, b^{[K]}$

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

Training a prediction function for a reconstruction task

Deep learning – *General framework*

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

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

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

Training a prediction function for a reconstruction task

Deep learning – *General framework*

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

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

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

- Evaluate: A properly trained network must satisfy

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

Variational approach versus Deep learning architecture

Variational approach

$$\mathrm{d}_{\boldsymbol{\Theta}}(\mathrm{z}_\ell) = \underset{\mathrm{x}\in\mathcal{H}}{\mathrm{Argmin}} \Big\{ \mathsf{F}(\mathrm{x}) := f(\mathrm{Ax}, \mathrm{z}_\ell) + g(\mathrm{Lx}) \Big\}$$

- ⊙ Data fidelity term: $f(\mathrm{A}\cdot, \mathrm{z}_i)$
- ⊙ Prior: $g(\mathrm{L}\cdot)$

$\Rightarrow$ $\boldsymbol{\Theta} = \{\mathrm{L}, K = \#\text{iter}\}$

Deep learning

$$\mathrm{d}_{\boldsymbol{\Theta}}(\mathrm{z}_\ell) = \eta^{[K]}\big(W^{[K]} \ldots \eta^{[1]}(W^{[1]}\mathrm{z}_\ell + b^{[1]}) \ldots + b^{[K]}\big)$$

- ⊙ Linear operators: $W^{[1]}, \mathrm{W}^{[2]}, \ldots, W^{[K]}$
- ⊙ Activation functions: $\eta^{[1]}, \eta^{[2]}, \ldots, \eta^{[K]}$
- ⊙ Biais vectors: $b^{[1]}, b^{[2]}, \ldots, b^{[K]}$

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

Synthesis formulation and proximal gradient descent: LISTA

➡ **Synthesis formulation:**

$$\boxed{\min_{\mathrm{x}} \frac{1}{2}\|\mathrm{Hx}-\mathrm{z}\|_2^2 + \lambda\|\mathrm{x}\|_1} \text{ where } \mathrm{H} = \mathrm{AL}^* \in \mathbb{R}^{N\times\tilde{N}}$$

➡ **Forward-backward iterations:**

$$\left|\ \mathrm{x}^{[k+1]} = \mathrm{prox}_{\tau\lambda\|\cdot\|_1}(\mathrm{x}^{[k]} - \tau\mathrm{H}^*(\mathrm{Hx}^{[k]} - \mathrm{z}))\right.$$

➡ **Reformulation:**

$$\left|\ \mathrm{x}^{[k+1]} = \mathrm{prox}_{\tau\lambda\|\cdot\|_1}((\mathrm{Id} - \tau\mathrm{H}^*\mathrm{H})\mathrm{x}^{[k]} + \tau\mathrm{H}^*\ \mathrm{z}))\right.$$

➡ **Layer network:** [Gregor, LeCun, 2010]

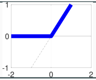
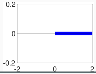
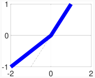
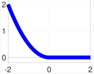
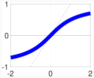
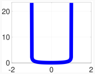
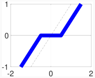
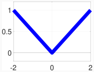
$$\mathrm{x}^{[k+1]} = \underbrace{\boxed{\mathrm{prox}_{\tau\lambda\|\cdot\|_1}}}_{\eta^{[k]}}\left(\underbrace{\boxed{\mathrm{Id} - \tau\mathrm{H}^*\mathrm{H}}}_{W^{[k]}}\ \mathrm{x}^{[k]} + \underbrace{\boxed{\tau\mathrm{H}^*\mathrm{z}}}_{b^{[k]}}\right)$$

Standard activation functions

➡ **Preliminary remarks** [Combettes, Pesquet, 2020]

- **Most of activation functions are proximity operator** :
  ReLU, Unimodal sigmoid, Softmax ...

Standard activation functions

Name	Activation $x \in \mathbb{R} \mapsto \varrho(x) = \mathrm{prox}_{\varphi}(x)$		Potential $x \in \mathbb{R} \mapsto \varphi(x)$	
ReLU	$\begin{cases} x & \text{if } x > 0 \\ 0 & \text{otherwise} \end{cases}$		$\iota_{[0,+\infty)}$	
Param. ReLU	$\begin{cases} x & \text{if } x > 0 \\ \lambda x & \text{otherwise} \end{cases}$		$\begin{cases} 0 & \text{if } x > 0 \\ \dfrac{(1/\lambda - 1)x^2}{2} & \text{otherwise} \end{cases}$	
Arctan.	$\dfrac{2\arctan(x)}{\pi}$		$\begin{cases} -\frac{2}{\pi} \ln\left(\cos\left(\frac{\pi x}{2}\right)\right) - \frac{x^2}{2} & \text{if } \lvert x \rvert < 1 \\ +\infty & \text{otherwise} \end{cases}$	
Soft thresh.	$\mathrm{sign}(x) \max\{\lvert x \rvert - \lambda, 0\}$		$\lambda \lvert x \rvert$	

Standard activation functions

➡ **Preliminary remarks** [Combettes, Pesquet, 2020]

- **Most of activation functions are proximity operator** : ReLU, Unimodal sigmoid, Softmax ...
- For $\mathrm{W}^{[k]}$ bounded linear operators and $\eta^{[k]}$ proximity operators, d_Θ model allows to derive tight Lipschitz bounds for feedforward neural networks in order to evaluate **robustness**.

Study case: Focus on denoising

Iterative scheme

Minimization problem : $\widehat{\mathrm{x}} = \arg\min_{\mathrm{x}} \frac{1}{2}\|\mathrm{x}-\mathrm{z}\|^2 + \|\mathrm{L}\mathrm{x}\|_1$

Dual reformulation: $\widehat{\mathrm{u}} \in \operatorname{Argmin}_{\mathrm{u}\in\mathcal{G}} \frac{1}{2}\|\mathrm{z}-\mathrm{L}^\top\mathrm{u}\|^2 + \iota_{\|\cdot\|_\infty\leq 1}(\mathrm{u})$

- Primal solution: $\widehat{\mathrm{x}} = \mathrm{z} - \mathrm{L}^\top\widehat{\mathrm{u}}$.
- Solution obtained with proximal gradient based procedure.
- Accelerated schemes (e.g., FISTA).

(F)ISTA in the dual

Minimization problem : $\widehat{\mathrm{x}} = \arg\min_{\mathrm{x}} \frac{1}{2}\|\mathrm{x}-\mathrm{z}\|^2 + \|\mathrm{Lx}\|_1$

Dual reformulation: $\widehat{\mathrm{u}} \in \operatorname*{Argmin}_{\mathrm{u}\in\mathcal{G}} \frac{1}{2}\|\mathrm{z}-\mathrm{L}^\top\mathrm{u}\|^2 + \iota_{\|\cdot\|_\infty\le 1}(\mathrm{u})$

(F)ISTA to solve dual reformulation:

Set $\mathrm{u}^{[0]} \in \mathbb{R}^{|\mathbb{F}|}$, and $\mathrm{v}^{[0]} \in \mathbb{R}^{|\mathbb{F}|}$.

$$\left\lfloor\begin{array}{ll} \mathrm{u}^{[k+1]} & = \operatorname{prox}_{\iota_{\|\cdot\|_\infty\le 1}}\Big((\mathrm{Id}-\tau_k\mathrm{LL}^\top)\mathrm{v}^{[k]} + \tau_k\mathrm{Lz}\Big) \\ \mathrm{v}^{[k+1]} & = (1+\alpha_k)\mathrm{u}^{[k+1]} - \alpha_k\mathrm{u}^{[k]} \end{array}\right.$$

Preliminary remarks:

- FISTA when $\alpha_k = \frac{\theta_k - 1}{\theta_{k+1}}$.
- ISTA when $\alpha_k \equiv 0$.
- (F)ISTA: $\widehat{\mathrm{x}} = \mathrm{z} - \mathrm{L}^\top\widehat{\mathrm{u}}$

(F)ISTA in the dual

(F)ISTA to solve dual reformulation:

Set $\mathrm{u}^{[0]} \in \mathbb{R}^{|\mathbb{F}|}$, and $\mathrm{v}^{[0]} \in \mathbb{R}^{|\mathbb{F}|}$.

$$\left\lfloor \begin{array}{ll} \mathrm{u}^{[k+1]} & = \mathrm{prox}_{\iota_{\|\cdot\|_\infty \leq 1}}\left((\mathrm{Id} - \tau_k \mathrm{L}\mathrm{L}^\top)\mathrm{v}^{[k]} + \tau_k \mathrm{L}\mathrm{z} \right) \\ \mathrm{v}^{[k+1]} & = (1+\alpha_k)\mathrm{u}^{[k+1]} - \alpha_k \mathrm{u}^{[k]} \end{array} \right.$$

Proposition : The proximity operator of the conjugate of the ℓ_1-norm scaled by parameter $\lambda > 0$ fits the HardTanh activation function,:

$$(\forall \mathrm{x} = (\mathrm{x}_i)_{1 \leq i \leq N}) \qquad \mathrm{P}_{\|\cdot\|_\infty \leq \lambda}(\mathrm{x}) = \mathrm{HardTanh}_\lambda(\mathrm{x}) = (\mathrm{p}_i)_{1 \leq i \leq N}$$

where

$$\mathrm{p}_i = \begin{cases} -\lambda & \text{if} \quad \mathrm{p}_i < -\lambda, \\ \lambda & \text{if} \quad \mathrm{p}_i > \lambda, \\ \mathrm{p}_i & \text{otherwise.} \end{cases}$$

(F)ISTA in the dual

(F)ISTA to solve dual reformulation:

Set $\mathrm{u}^{[0]} \in \mathbb{R}^{|\mathbb{F}|}$, and $\mathrm{v}^{[0]} \in \mathbb{R}^{|\mathbb{F}|}$.

$$\left\lfloor \begin{aligned} \mathrm{u}^{[k+1]} &= \boxed{\mathrm{HardTanh}_1}\Big((\mathrm{Id} - \tau_k \mathrm{L}\mathrm{L}^\top)\mathrm{v}^{[k]} + \tau_k \mathrm{L}\mathrm{z}\Big) \\ \mathrm{v}^{[k+1]} &= (1+\alpha_k)\mathrm{u}^{[k+1]} - \alpha_k \mathrm{u}^{[k]} \end{aligned} \right.$$

Proposition : The proximity operator of the conjugate of the ℓ_1-norm scaled by parameter $\lambda > 0$ fits the HardTanh activation function,:

$$(\forall \mathrm{x} = (\mathrm{x}_i)_{1\le i\le N}) \qquad \mathrm{P}_{\|\cdot\|_\infty \le \lambda}(\mathrm{x}) = \mathrm{HardTanh}_\lambda(\mathrm{x}) = (\mathrm{p}_i)_{1\le i\le N}$$

where

$$\mathrm{p}_i = \begin{cases} -\lambda & \text{if} \quad \mathrm{p}_i < -\lambda, \\ \lambda & \text{if} \quad \mathrm{p}_i > \lambda, \\ \mathrm{p}_i & \text{otherwise.} \end{cases}$$

(F)ISTA in the dual

(F)ISTA to solve dual reformulation:

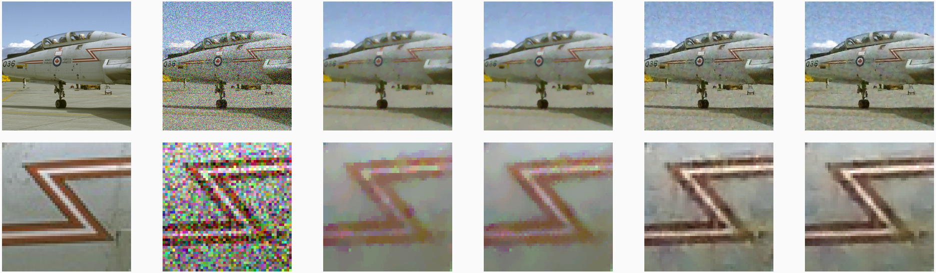
Set $\mathrm{u}^{[0]} \in \mathbb{R}^{|\mathbb{F}|}$, and $\mathrm{v}^{[0]} \in \mathbb{R}^{|\mathbb{F}|}$.

$$\left\lfloor \begin{array}{ll} \mathrm{u}^{[k+1]} & = \mathrm{HardTanh}_1\Big((\mathrm{Id} - \tau_k \mathrm{L}\mathrm{L}^\top)\mathrm{v}^{[k]} + \tau_k \mathrm{L}\mathrm{z}\Big) \\ \mathrm{v}^{[k+1]} & = (1+\alpha_k)\mathrm{u}^{[k+1]} - \alpha_k \mathrm{u}^{[k]} \end{array} \right.$$

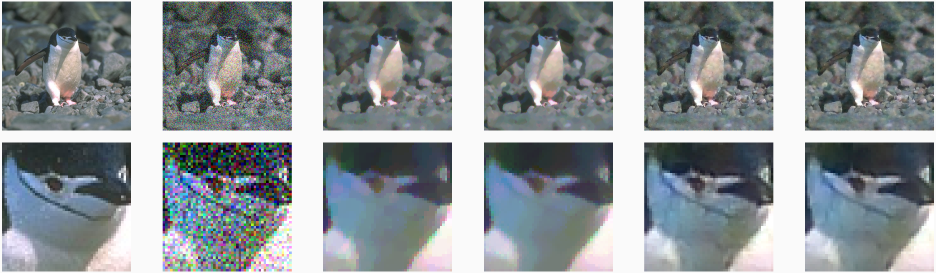
➡ **Unfolded (F)ISTA**:

$$\begin{bmatrix} \mathrm{u}^{[k]} \\ \mathrm{u}^{[k+1]} \end{bmatrix} = \underbrace{\boxed{\begin{Bmatrix} \mathrm{Id}_{|\mathbb{F}|} \\ \mathrm{HardTanh}_1 \end{Bmatrix}}}_{\eta^{[k]}} \Bigg(\underbrace{\boxed{\begin{bmatrix} 0 & \mathrm{Id}_{|\mathbb{F}|} \\ -\alpha_{k-1}(\mathrm{Id}_{|\mathbb{F}|} - \mathrm{L}_1^{[k]}\mathrm{L}_2^{[k]}) & (1+\alpha_{k-1})(\mathrm{Id}_{|\mathbb{F}|} - \mathrm{L}_1^{[k]}\mathrm{L}_2^{[k]}) \end{bmatrix}}}_{\mathrm{W}^{[k]}} \begin{bmatrix} \mathrm{u}^{[k-1]} \\ \mathrm{u}^{[k]} \end{bmatrix} + \underbrace{\boxed{\begin{bmatrix} 0 \\ \mathrm{L}_1^{[k]}\mathrm{z}_l \end{bmatrix}}}_{\mathrm{b}^{[k]}}$$

Original	Noisy	TV	NL-TV	DnCNN	Proposed

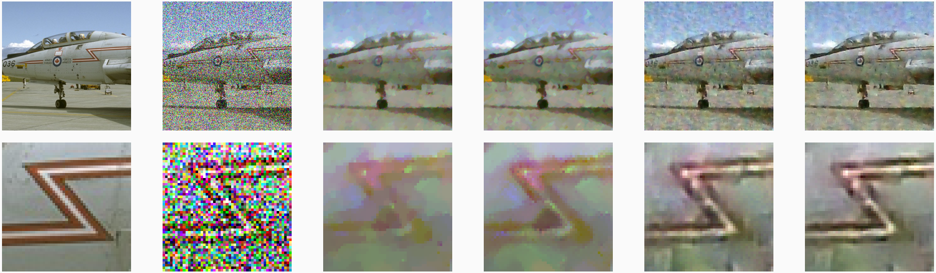


PSNR/SSIM	14.1/0.25	26.0/0.84	26.6/0.85	27.9/0.86	**28.2/0.87**

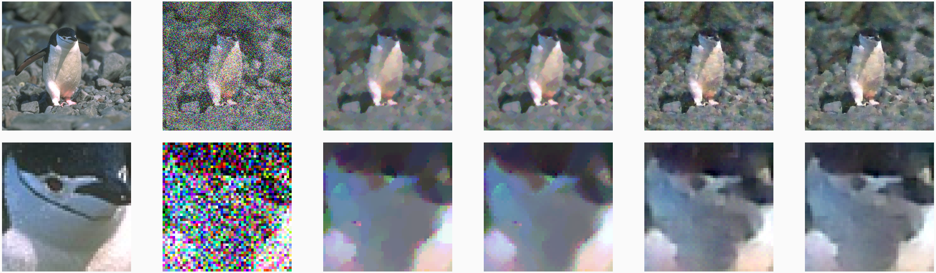


PSNR/SSIM	14.1/0.13	26.0/0.76	27.7/0.79	28.5/0.79	**28.8**/**0.81**

Original	Noisy	TV	NL-TV	DnCNN	Proposed



PSNR/SSIM	8.13/0.09	23.6/0.76	24.0/0.76	24.4/0.76	**25.2/0.80**



PSNR/SSIM	8.14/0.043	24.5/0.64	25.1/0.65	25.4/0.65	**25.9/0.70**

D(i)FB algorithm

Objective: $$\widehat{\mathrm{x}} = \operatorname*{argmin}_{\mathrm{x}\in\mathcal{H}} \Big\{ \mathsf{F}(\mathrm{x}) = \tfrac{1}{2}\|\mathrm{x}-\mathrm{z}\|_2^2 + g(\mathrm{L}\mathrm{x}) + \iota_C(\mathrm{x}) \Big\}$$

- $C \subset \mathcal{H}$ is a closed, convex, non-empty.
- $\mathrm{L}\colon \mathcal{H} \to \mathcal{G}$ and g $\in \Gamma_0(\mathcal{G})$

D(i)FB algorithm

OBJECTIVE: $\widehat{\mathrm{x}} = \underset{\mathrm{x}\in\mathcal{H}}{\mathsf{argmin}} \Big\{ \mathsf{F}(\mathrm{x}) = \frac{1}{2}\|\mathrm{x}-\mathrm{z}\|_2^2 + g(\mathrm{Lx}) + \iota_C(\mathrm{x}) \Big\}$

- $C \subset \mathcal{H}$ is a closed, convex, non-empty.
- $\mathrm{L}\colon \mathcal{H} \to \mathcal{G}$ and g $\in \Gamma_0(\mathcal{G})$

ALGORITHM: Let $\mathrm{v}^{[0]} \in \mathcal{G}$,

For $k = 0, 1, \ldots$

$$
\left\lfloor
\begin{array}{l}
\mathrm{u}^{[k+1]} = \mathrm{prox}_{\tau_k (\nu g)^*}\Big(\mathrm{v}^{[k]} + \tau_k \mathrm{L} \mathrm{P}_C(\mathrm{z} - \mathrm{L}^\top \mathrm{v}^{[k]})\Big) \\
\mathrm{v}^{[k+1]} = (1+\alpha_k)\mathrm{u}^{[k+1]} - \alpha_k \mathrm{u}^{[k]}
\end{array}
\right.
$$

D(i)FB algorithm

OBJECTIVE: $\widehat{\mathrm{x}} = \underset{\mathrm{x}\in\mathcal{H}}{\mathrm{argmin}} \left\{ \mathsf{F}(\mathrm{x}) = \frac{1}{2}\|\mathrm{x}-\mathrm{z}\|_2^2 + g(\mathrm{Lx}) + \iota_C(\mathrm{x}) \right\}$

- $C \subset \mathcal{H}$ is a closed, convex, non-empty.
- $\mathrm{L}\colon \mathcal{H} \to \mathcal{G}$ and $\mathrm{g} \in \Gamma_0(\mathcal{G})$

ALGORITHM: Let $\mathrm{v}^{[0]} \in \mathcal{G}$,

For $k = 0, 1, \dots$

$$\left\lfloor \begin{array}{l} \mathrm{u}^{[k+1]} = \mathrm{prox}_{\tau_k(\nu g)^*}\left(\mathrm{v}^{[k]} + \tau_k \mathrm{L} \mathrm{P}_C(\mathrm{z} - \mathrm{L}^\top \mathrm{v}^{[k]})\right) \\ \mathrm{v}^{[k+1]} = (1+\alpha_k)\mathrm{u}^{[k+1]} - \alpha_k \mathrm{u}^{[k]} \end{array} \right.$$

THEOREM: Assume that one of the following conditions is satisfied.

- (DFB): $\forall k \in$, $\tau_k \in (0, 2/\|\mathrm{L}\|_S^2)$, and $\alpha_k = 0$.
- (DIFB): $\forall k \in$, $\tau_k \in (0, 1/\|\mathrm{L}\|_S^2)$, $\alpha_k = \frac{\theta_k - 1}{\theta_{k+1}}$ with $\theta_k = \frac{k+a}{a}$ and $a > 2$.

Then we have $\widehat{\mathrm{x}} = \lim_{k\to\infty} \mathrm{P}_C(\mathrm{z} - \mathrm{L}^\top \mathrm{u}^{[k]})$.

(Sc)CP algorithm

OBJECTIVE: $\quad \widehat{\mathrm{x}} = \underset{\mathrm{x}\in\mathcal{H}}{\mathsf{argmin}}\Big\{\mathsf{F}(\mathrm{x}) = \frac{1}{2}\|\mathrm{x}-\mathrm{z}\|_2^2 + g(\mathrm{Lx}) + \iota_C(\mathrm{x})\Big\}$

- $C \subset \mathcal{H}$ is a closed, convex, non-empty.
- $\mathrm{L}\colon \mathcal{H} \to \mathcal{G}$ and $\mathsf{g} \in \Gamma_0(\mathcal{G})$

(Sc)CP algorithm

OBJECTIVE: $\widehat{\mathrm{x}} = \underset{\mathrm{x}\in\mathcal{H}}{\mathsf{argmin}}\left\{\mathsf{F}(\mathrm{x}) = \frac{1}{2}\|\mathrm{x}-\mathrm{z}\|_2^2 + g(\mathrm{Lx}) + \iota_C(\mathrm{x})\right\}$

- $C \subset \mathcal{H}$ is a closed, convex, non-empty.
- $\mathrm{L}\colon \mathcal{H} \to \mathcal{G}$ and $\mathsf{g} \in \Gamma_0(\mathcal{G})$

ALGORITHM: Let $\mathrm{x}^{[0]} \in \mathcal{H}$ and $\mathbf{u}^{[0]} \in \mathcal{G}$.

For $k = 0, 1, \dots$

$$\left\lfloor \begin{array}{l} \mathrm{x}^{[k+1]} = \mathrm{P}_C\left(\frac{\mu_k}{1+\mu_k}(\mathrm{z} - \mathrm{L}^\top \mathrm{u}^{[k]}) + \frac{1}{1+\mu_k}\mathrm{x}^{[k]}\right) \\ \mathrm{u}^{[k+1]} = \mathrm{prox}_{\tau_k(\nu g)^*}\left(\mathrm{u}^{[k]} + \tau_k \mathrm{L}\Big((1+\alpha_k)\mathrm{x}^{[k+1]} - \alpha_k \mathrm{x}^{[k]}\Big)\right) \end{array}\right.$$

(Sc)CP algorithm

OBJECTIVE: $\widehat{\mathrm{x}} = \underset{\mathrm{x}\in\mathcal{H}}{\mathrm{argmin}} \Big\{ \mathsf{F}(\mathrm{x}) = \frac{1}{2}\|\mathrm{x}-\mathrm{z}\|_2^2 + g(\mathrm{Lx}) + \iota_C(\mathrm{x}) \Big\}$

- $C \subset \mathcal{H}$ is a closed, convex, non-empty.
- $\mathrm{L}\colon \mathcal{H} \to \mathcal{G}$ and $\mathsf{g} \in \Gamma_0(\mathcal{G})$

ALGORITHM: Let $\mathrm{x}^{[0]} \in \mathcal{H}$ and $\mathbf{u}^{[0]} \in \mathcal{G}$.

For $k = 0, 1, \ldots$

$$
\left\lfloor
\begin{aligned}
\mathrm{x}^{[k+1]} &= \mathrm{P}_C\left(\tfrac{\mu_k}{1+\mu_k}(\mathrm{z} - \mathrm{L}^\top \mathrm{u}^{[k]}) + \tfrac{1}{1+\mu_k}\mathrm{x}^{[k]} \right) \\
\mathrm{u}^{[k+1]} &= \mathrm{prox}_{\tau_k(\nu g)^*}\left(\mathrm{u}^{[k]} + \tau_k \mathrm{L}\Big((1+\alpha_k)\mathrm{x}^{[k+1]} - \alpha_k \mathrm{x}^{[k]} \Big) \right)
\end{aligned}
\right.
$$

THEOREM: Assume that one of the following conditions is satisfied.

- (CP): $\tau_k \mu_k \|\mathrm{L}\|_S^2 < 1$, and $\alpha_k = 1$.
- (ScCP): $\alpha_k = \sqrt{1+2\mu_k}^{\,-1}$, $\mu_{k+1} = \alpha_k \mu_k$, $\tau_{k+1} = \tau_k \alpha_k^{-1}$ with $\mu_0 \tau_0 \|\mathrm{L}\|_S^2 \leq 1$.

Then we have $\widehat{\mathrm{x}} = \lim_{k\to\infty} \mathrm{x}^{[k]}$.

S(c)CP to D(i)FB

$$\textsc{Objective:}\quad \widehat{\mathrm{x}} = \underset{\mathrm{x}\in\mathcal{H}}{\operatorname{argmin}}\Big\{\mathsf{F}(\mathrm{x}) = \tfrac{1}{2}\|\mathrm{x}-\mathrm{z}\|_2^2 + g(\mathrm{L}\mathrm{x}) + \iota_C(\mathrm{x})\Big\}$$

ALGORITHM: For $k = 0, 1, \dots$

$$\left\lfloor\begin{array}{l} \mathrm{x}^{[k+1]} = \mathrm{P}_C\left(\frac{\mu_k}{1+\mu_k}(\mathrm{z}-\mathrm{L}^\top\mathrm{u}^{[k]}) + \frac{1}{1+\mu_k}\mathrm{x}^{[k]}\right) \\ \mathrm{u}^{[k+1]} = \mathrm{prox}_{\tau_k(\nu g)^*}\left(\mathrm{u}^{[k]} + \tau_k\mathrm{L}\Big((1+\alpha_k)\mathrm{x}^{[k+1]} - \alpha_k\mathrm{x}_k\Big)\right) \end{array}\right.$$

☛ **S(c)CP:** Starting point.

S(c)CP to D(i)FB

Objective: $\widehat{\mathrm{x}} = \operatorname*{argmin}_{\mathrm{x}\in\mathcal{H}} \Big\{ \mathsf{F}(\mathrm{x}) = \frac{1}{2}\|\mathrm{x}-\mathrm{z}\|_2^2 + g(\mathrm{Lx}) + \iota_C(\mathrm{x}) \Big\}$

Algorithm: For $k = 0, 1, \ldots$

$$\left\lfloor \begin{array}{l} \mathrm{x}^{[k+1]} = \mathrm{P}_C\left(\frac{\mu_k}{1+\mu_k}(\mathrm{z} - \mathrm{L}^\top \mathrm{u}^{[k]}) + \frac{1}{1+\mu_k}\mathrm{x}^{[k]}\right) \\ \mathrm{u}^{[k+1]} = \mathrm{prox}_{\tau_k(\nu g)^*}\left(\mathrm{u}^{[k]} + \tau_k \mathrm{L}\Big((1+\alpha_k)\mathrm{x}^{[k+1]} - \alpha_k \mathrm{x}_k\Big)\right) \end{array} \right.$$

- **S(c)CP:** Starting point.
- **Arrow-Hurwicz iterations:** $\alpha_k \equiv 0$.

S(c)CP to D(i)FB

$$\textsc{Objective:}\quad \widehat{x} = \underset{x\in\mathcal{H}}{\mathsf{argmin}}\Big\{\mathsf{F}(x) = \tfrac{1}{2}\|x - z\|_2^2 + g(\mathrm{L}x) + \iota_C(x)\Big\}$$

$\textsc{Algorithm:}$ For $k = 0, 1, \dots$

$$\left\lfloor \begin{array}{l} x^{[k+1]} = \mathrm{P}_C\left(\frac{\mu_k}{1+\mu_k}(z - \mathrm{L}^\top u^{[k]}) + \frac{1}{1+\mu_k}x^{[k]}\right) \\ u^{[k+1]} = \mathrm{prox}_{\tau_k(\nu g)^*}\left(u^{[k]} + \tau_k \mathrm{L}x^{[k+1]}\right) \end{array} \right.$$

- **S(c)CP:** Starting point.
- **Arrow-Hurwicz iterations:** $\alpha_k \equiv 0$.

S(c)CP to D(i)FB

OBJECTIVE: $\widehat{\mathrm{x}} = \underset{\mathrm{x}\in\mathcal{H}}{\operatorname{argmin}} \left\{ \mathsf{F}(\mathrm{x}) = \frac{1}{2}\|\mathrm{x}-\mathrm{z}\|_2^2 + g(\mathrm{Lx}) + \iota_C(\mathrm{x}) \right\}$

ALGORITHM: For $k = 0, 1, \ldots$

$$
\begin{array}{l}
\mathrm{x}^{[k+1]} = \mathrm{P}_C\left(\frac{\mu_k}{1+\mu_k}(\mathrm{z} - \mathrm{L}^\top \mathrm{u}^{[k]}) + \frac{1}{1+\mu_k}\mathrm{x}^{[k]}\right) \\
\mathrm{u}^{[k+1]} = \mathrm{prox}_{\tau_k(\nu g)^*}\left(\mathrm{u}^{[k]} + \tau_k \mathrm{Lx}^{[k+1]}\right)
\end{array}
$$

- **S(c)CP:** Starting point.
- **Arrow-Hurwicz iterations:** $\alpha_k \equiv 0$.
- **DFB:** $\mu_k \to +\infty$.

S(c)CP to D(i)FB

OBJECTIVE: $\widehat{\mathrm{x}} = \underset{\mathrm{x}\in\mathcal{H}}{\operatorname{argmin}} \Big\{ \mathsf{F}(\mathrm{x}) = \frac{1}{2}\|\mathrm{x}-\mathrm{z}\|_2^2 + g(\mathrm{L}\mathrm{x}) + \iota_C(\mathrm{x}) \Big\}$

ALGORITHM: For $k = 0, 1, \ldots$

$$
\left\lfloor
\begin{array}{l}
\mathrm{x}^{[k+1]} = \mathrm{P}_C\left(\mathrm{z} - \mathrm{L}^\top \mathrm{u}^{[k]}\right) \\
\mathrm{u}^{[k+1]} = \mathrm{prox}_{\tau_k(\nu g)^*}\left(\mathrm{u}^{[k]} + \tau_k \mathrm{L}\mathrm{x}^{[k+1]}\right)
\end{array}
\right.
$$

- **S(c)CP:** Starting point.
- **Arrow-Hurwicz iterations:** $\alpha_k \equiv 0$.
- **DFB:** $\mu_k \to +\infty$.

S(c)CP to D(i)FB

OBJECTIVE: $\widehat{\mathrm{x}} = \underset{\mathrm{x}\in\mathcal{H}}{\operatorname{argmin}}\Big\{\mathsf{F}(\mathrm{x}) = \frac{1}{2}\|\mathrm{x}-\mathrm{z}\|_2^2 + g(\mathrm{Lx}) + \iota_C(\mathrm{x})\Big\}$

ALGORITHM: For $k = 0, 1, \ldots$

$$\left\lfloor \begin{array}{l} \mathrm{x}^{[k+1]} = \mathrm{P}_C\left(\mathrm{z} - \mathrm{L}^\top \mathrm{u}^{[k]}\right) \\ \mathrm{u}^{[k+1]} = \mathrm{prox}_{\tau_k(\nu g)^*}\left(\mathrm{u}^{[k]} + \tau_k \mathrm{L}\mathrm{x}^{[k+1]}\right) \end{array} \right.$$

- **S(c)CP:** Starting point.
- **Arrow-Hurwicz iterations:** $\alpha_k \equiv 0$.
- **DFB:** $\mu_k \to +\infty$.
- **DiFB:** Inertia step on the dual variable.

S(c)CP to D(i)FB

OBJECTIVE: $\widehat{\mathrm{x}} = \underset{\mathrm{x}\in\mathcal{H}}{\mathrm{argmin}} \Big\{ \mathsf{F}(\mathrm{x}) = \frac{1}{2}\|\mathrm{x}-\mathrm{z}\|_2^2 + g(\mathrm{L}\mathrm{x}) + \iota_C(\mathrm{x}) \Big\}$

ALGORITHM: For $k = 0, 1, \ldots$

$$
\begin{aligned}
&\mathrm{x}^{[k+1]} = \mathrm{P}_C\left(\mathrm{z} - \mathrm{L}^\top \mathrm{v}^{[k]}\right)\\
&\mathrm{u}^{[k+1]} = \mathrm{prox}_{\tau_k(\nu g)^*}\left(\mathrm{u}^{[k]} + \tau_k \mathrm{L}\mathrm{x}^{[k+1]}\right)\\
&\mathrm{v}^{[k+1]} = (1+\rho_k)\mathrm{u}^{[k+1]} - \rho_k \mathrm{u}^{[k]}
\end{aligned}
$$

- **S(c)CP:** Starting point.
- **Arrow-Hurwicz iterations:** $\alpha_k \equiv 0$.
- **DFB:** $\mu_k \to +\infty$.
- **DiFB:** Inertia step on the dual variable.

Arrow-Hurwicz building block

ITERATION: Arrow-Hurwicz iteration can be written as:

$$\begin{aligned} \mathrm{d}_{\mathrm{z},\nu,\Theta_k} : \quad & \mathcal{H} \times \mathcal{G} \to \mathcal{H} \\ & (\mathrm{x}^{[k]}, \mathrm{u}^{[k]}) \mapsto \mathrm{d}_{\mathrm{z},\Theta_{k,\mathcal{P}},\mathcal{P}}(\mathrm{x}, \mathrm{d}_{\Theta_{k,\mathcal{D}},\mathcal{D}}(\mathrm{x}^{[k]}, \mathrm{u}^{[k]})) \end{aligned}$$

with

$$\mathrm{d}_{\nu,\Theta_{k,\mathcal{D}},\mathcal{D}}(\mathrm{x}, \mathrm{u}) = \mathrm{prox}_{\tau_k(\nu g)^*}\left(\tau_k \mathrm{L}\mathrm{x} + \mathrm{u}\right),$$

$$\mathrm{d}_{\mathrm{z},\Theta_{k,\mathcal{P}},\mathcal{P}}(\mathrm{x}, \mathrm{u}) = \mathrm{P}_C\left(\frac{1}{1+\mu_k}\mathrm{x} - \frac{\mu_k}{1+\mu_k}\mathrm{L}^\top \mathrm{u} + \frac{\mu_k}{1+\mu_k}\mathrm{z}\right)$$

DEEP LEARNING NOTATION:

$$\mathrm{d}_{\boldsymbol{\Theta}} = \eta^{[K]}\big(W^{[K]} \dots \eta^{[1]}(W^{[1]} \cdot + b^{[1]}) \dots + b^{[K]}\big)$$

Deep Arrow-Hurwicz building block

LAYER: Arrow-Hurwicz layer can be written as:

$$\begin{aligned} \mathrm{d}_{\mathrm{z},\nu,\Theta_k} \colon \quad \mathcal{H} \times \mathcal{G} \; &\to \mathcal{H} \\ (\mathrm{x}^{[k]}, \mathrm{u}^{[k]}) &\mapsto \mathrm{d}_{\mathrm{z},\Theta_{k,\mathcal{P}},\mathcal{P}}(\mathrm{x}^{[k]}, \mathrm{d}_{\Theta_{k,\mathcal{D}},\mathcal{D}}(\mathrm{x}^{[k]}, \mathrm{u}^{[k]})), \end{aligned}$$

with

$$\mathrm{d}_{\nu,\Theta_{k,\mathcal{D}},\mathcal{D}}(\mathrm{x}, \mathrm{u}) = \mathrm{prox}_{\tau_k(\nu g)^*}\left(\tau_k \mathrm{L}_{k,\mathcal{D}}\mathrm{x} + \mathrm{u}\right),$$

$$\mathrm{d}_{\mathrm{z},\Theta_{k,\mathcal{P}},\mathcal{P}}(\mathrm{x}, \mathrm{u}) = \mathrm{P}_C\left(\frac{1}{1+\mu_k}\mathrm{x} - \frac{\mu_k}{1+\mu_k}\mathrm{L}_{k,\mathcal{P}}^{\top}\mathrm{u} + \frac{\mu_k}{1+\mu_k}\mathrm{z}\right),$$

Deep Arrow-Hurwicz building block + skip connections

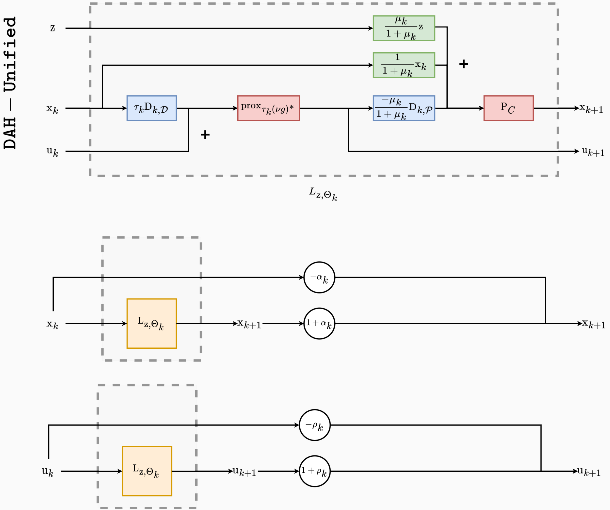


Figure 1: Architecture of the proposed `DAH-Unified` block for the k-th layer. `Inertial step` for ScCP (top) and DiFB (bottom).

Variations on the proposed architecture

	Θ_k	Comments
DDFB-LFO	$\mathrm{L}_{k,\mathcal{P}}$, $\mathrm{L}_{k,\mathcal{D}}$	absorb τ_k in $\mathrm{L}_{k,\mathcal{D}}$
DDFB-LNO	$\mathrm{L}_{k,\mathcal{P}} = \mathrm{L}_{k,\mathcal{D}}^\top$	define $\tau_k = 1.99\|\mathrm{L}_k\|^{-2}$

Variations on the proposed architecture

	Θ_k	Comments
DDFB-LFO	$\mathrm{L}_{k,\mathcal{P}}$, $\mathrm{L}_{k,\mathcal{D}}$	absorb τ_k in $\mathrm{L}_{k,\mathcal{D}}$
DDFB-LNO	$\mathrm{L}_{k,\mathcal{P}} = \mathrm{L}_{k,\mathcal{D}}^\top$	define $\tau_k = 1.99\|\mathrm{L}_k\|^{-2}$
DCP-LFO	$\mathrm{L}_{k,\mathcal{P}}$, $\mathrm{L}_{k,\mathcal{D}}$, μ	learn $\mu = \mu_0 = \cdots = \mu_K$, and absorb τ_k in $\mathrm{L}_{k,\mathcal{D}}$
DCP-LNO	$\mathrm{L}_{k,\mathcal{P}} = \mathrm{L}_{k,\mathcal{D}}^\top$, μ	learn $\mu = \mu_0 = \cdots = \mu_K$, and fix $\tau_k = 0.99\mu^{-1}\|\mathrm{L}_k\|^{-2}$

Variations on the proposed architecture

	Θ_k	Comments
DDFB-LFO	$\mathrm{L}_{k,\mathcal{P}}$, $\mathrm{L}_{k,\mathcal{D}}$	absorb τ_k in $\mathrm{L}_{k,\mathcal{D}}$
DDiFB-LFO	$\mathrm{L}_{k,\mathcal{P}}$, $\mathrm{L}_{k,\mathcal{D}}$, α_k	fix α_k, and absorb τ_k in $\mathrm{L}_{k,\mathcal{D}}$
DDFB-LNO	$\mathrm{L}_{k,\mathcal{P}} = \mathrm{L}_{k,\mathcal{D}}^\top$	define $\tau_k = 1.99\|\mathrm{L}_k\|^{-2}$
DDiFB-LNO	$\mathrm{L}_{k,\mathcal{P}} = \mathrm{L}_{k,\mathcal{D}}^\top$	fix $\alpha_k = \frac{t_k-1}{t_{k+1}}$, $t_{k+1} = \frac{k+a-1}{a}$, $a > 2$, and $\tau_k = 0.99\|\mathrm{L}_k\|^{-2}$
DCP-LFO	$\mathrm{L}_{k,\mathcal{P}}$, $\mathrm{L}_{k,\mathcal{D}}$, μ	learn $\mu = \mu_0 = \cdots = \mu_K$, and absorb τ_k in $\mathrm{L}_{k,\mathcal{D}}$
DCP-LNO	$\mathrm{L}_{k,\mathcal{P}} = \mathrm{L}_{k,\mathcal{D}}^\top$, μ	learn $\mu = \mu_0 = \cdots = \mu_K$, and fix $\tau_k = 0.99\mu^{-1}\|\mathrm{L}_k\|^{-2}$

Variations on the proposed architecture

	Θ_k	Comments
DDFB-LFO	$\mathrm{L}_{k,\mathcal{P}}$, $\mathrm{L}_{k,\mathcal{D}}$	absorb τ_k in $\mathrm{L}_{k,\mathcal{D}}$
	$\mathrm{L}_{k,\mathcal{P}}$, $\mathrm{L}_{k,\mathcal{D}}$, α_k	fix α_k, and absorb τ_k in $\mathrm{L}_{k,\mathcal{D}}$
DDFB-LNO	$\mathrm{L}_{k,\mathcal{P}} = \mathrm{L}_{k,\mathcal{D}}^\top$	define $\tau_k = 1.99\|\mathrm{L}_k\|^{-2}$
DDiFB-LNO	$\mathrm{L}_{k,\mathcal{P}} = \mathrm{L}_{k,\mathcal{D}}^\top$	fix $\alpha_k = \frac{t_k - 1}{t_{k+1}}$, $t_{k+1} = \frac{k+a-1}{a}$, $a > 2$, and $\tau_k = 0.99\|\mathrm{L}_k\|^{-2}$
DCP-LFO	$\mathrm{L}_{k,\mathcal{P}}$, $\mathrm{L}_{k,\mathcal{D}}$, μ	learn $\mu = \mu_0 = \cdots = \mu_K$, and absorb τ_k in $\mathrm{L}_{k,\mathcal{D}}$
DScCP-LFO	$\mathrm{L}_{k,\mathcal{P}}$, $\mathrm{L}_{k,\mathcal{D}}$, μ_0	learn μ_0, absorb τ_k in $\mathrm{L}_{k,\mathcal{D}}$, and fix $\alpha_k = (1+2\mu_k)^{-1/2}$, and $\mu_{k+1} = \alpha_k \mu_k$
DCP-LNO	$\mathrm{L}_{k,\mathcal{P}} = \mathrm{L}_{k,\mathcal{D}}^\top$, μ	learn $\mu = \mu_0 = \cdots = \mu_K$, and fix $\tau_k = 0.99\mu^{-1}\|\mathrm{L}_k\|^{-2}$
DScCP-LNO	$\mathrm{L}_{k,\mathcal{P}} = \mathrm{L}_{k,\mathcal{D}}^\top$, μ_k	learn μ_k, and fix $\alpha_k = (1+2\mu_k)^{-1/2}$, and $\tau_k = 0.99\mu_k^{-1}\|\mathrm{L}_k\|^{-2}$

Limit case for deep unfolded NNs

[Le, Repetti, Pustelnik, 2023]

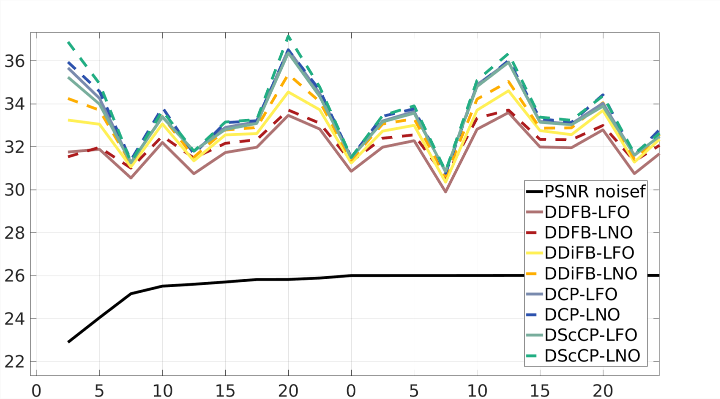
We consider the unfolded NNs DD(i)FB and D(Sc)CP. Assume that, for every $k \in \{1, \dots, K\}$, $\mathrm{L}_{k,\mathcal{D}} = \mathrm{L}$ and $\mathrm{L}_{k,\mathcal{P}} = \mathrm{L}^\top$, for $\mathrm{L} \colon \mathbb{R}^N \to \mathbb{R}^{|\mathbb{F}|}$. In addition, for each architecture, we further assume that, for every $k \in \{1, \dots, K\}$,

- DDFB: $\tau_k \in (0, 2/\|\mathrm{L}\|_S^2)$.
- DDiFB: $\tau_k \in (0, 1/\|\mathrm{L}\|_S^2)$ and $\rho_k = \frac{t_k - 1}{t_{k+1}}$ with $t_k = \frac{k+a-1}{a}$ and $a > 2$.
- DCP: $(\tau_k, \mu_k) \in (0, +\infty)^2$ such that $\tau_k \mu_k \|\mathrm{L}\|_S^2 < 1$.
- DScCP: $\alpha_k = (1 + 2\mu_k)^{-1/2}$, $\mu_{k+1} = \alpha_k \mu_k$, and $\tau_{k+1} = \tau_k \alpha_k^{-1}$ with $\tau_0 \mu_0 \|\mathrm{L}\|_S^2 \leq 1$.

Then, we have $\mathrm{x}_K \to \widehat{\mathrm{x}}$ when $K \to +\infty$, where x_K is the output of either of the unfolded NNs DD(i)FB or D(Sc)CP, and $\widehat{\mathrm{x}}$ is a solution to

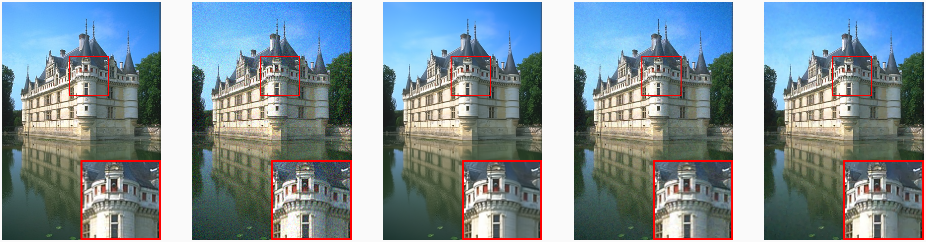
$$\min_{\mathrm{x} \in \mathcal{H}} \frac{1}{2} \|\mathrm{x} - \mathrm{z}\|_2^2 + g(\mathrm{L}\mathrm{x}) + \iota_C(\mathrm{x}).$$

Denoising performance



Denoising performance.

PSNR (with $(K, J) = (20, 64)$), for 20 images of BSDS500 validation set, degraded with noise level $\delta = 0.05$.



	Noisy	**DRUnet**	DDFB-LNO	DScCP-LNO
	26.03dB	**35.81dB**	32.81dB	34.74dB

Denoising performance on Gaussian noise. Example of denoised images (and PSNR values) for Gaussian noise $\delta = 0.05$ obtained with DRUnet and the proposed DDFB-LNO and DScCP-LNO, with $(K, J) = (20, 64)$.

Complexity of the models

		Time (msec)	$\|\Theta\|$	FLOPs ($\times 10^3$ G)
BM3D		$13{\times}10^3 \pm 317$	–	–
DRUnet		96 ± 21	$32,640,960$	137.24
LNO	DDFB	3 ± 1.5	$34,560$	2.26
	DDiFB	3 ± 0.5	$34,560$	
	DDCP	6 ± 1	$34,561$	
	DDScCP	7 ± 1	$34,580$	
LFO	DDFB	4 ± 17	$69,120$	2.26
	DDiFB	5 ± 15	$69,121$	
	DDCP	7 ± 14	$69,121$	
	DDScCP	9 ± 15	$69,160$	

NN robustness

☛ Given an input z and a perturbation ϵ, the error on the output can be upper bounded :

$$\|f_\Theta(\mathrm{z}+\epsilon) - f_\Theta(\mathrm{z})\| \leq \chi\|\epsilon\|.$$

where χ certificated of the robustness.

☛ [Combettes, Pesquet, 2020]: χ can be upper bounded by:

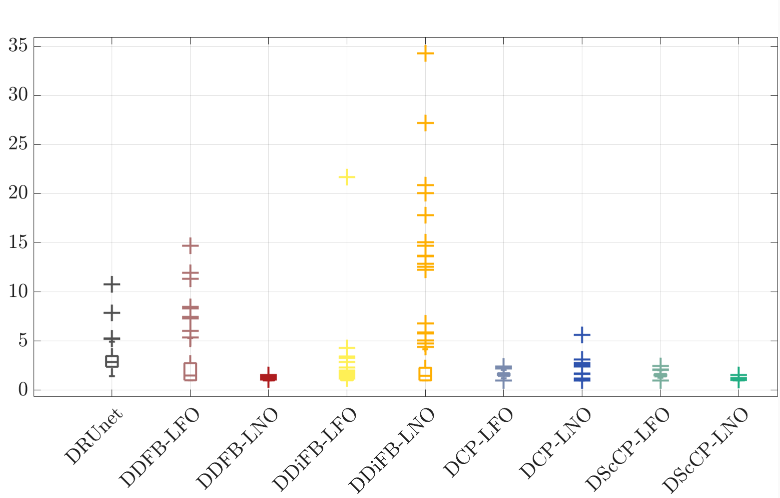
$$\chi \leq \prod_{k=1}^{K} \Big(\|W_{k,\mathcal{P}}\|_S \times \|W_{k,\mathcal{D}}\|_S \Big).$$

☛ [Pesquet, Repetti, Terris, Wiaux, 2020]: tighter bound by Lipschitz continuity:

$$\chi \approx \max_{(\mathrm{z}_s)_{s\in\mathbb{I}}} \| \mathrm{J}\, f_\Theta(\mathrm{z}_s)\|_S.$$

where J denotes the Jacobian operator.

NN robustness



Distribution of $(\| \mathrm{J}\, f_\Theta(\mathrm{z}_s)\|_S)_{s\in\mathbb{J}}$ for 100 images extracted from BSDS500 validation dataset $\mathbb{J}$, for the proposed PNNs and DRUnet.

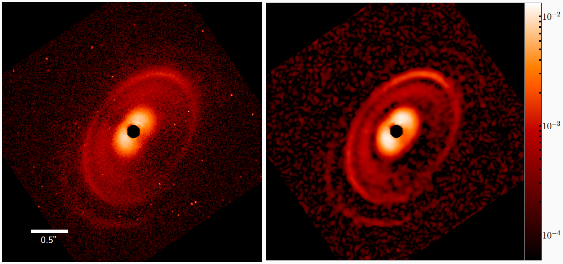
High contrast image reconstruction

Context: Image restoration

➡ **Data**: $\mathrm{z} \in \mathbb{R}^M$ degraded version of an original image $\overline{\mathrm{x}} \in \mathbb{R}^N$:

$$\boxed{\mathrm{z} = \mathrm{A}\overline{\mathrm{x}} + \mathbf{w}}$$

- $\mathrm{A} \in \mathbb{R}^{M \times N}$: linear degradation (e.g. a blur)
- $\mathbf{w}$: noise (e.g. Gaussian noise)



SPHERE-IRDIS

Context: Image restoration (astronomy context)

- **Studying circumstellar environments**: crucial for understanding exoplanets and stellar systems.
- **High contrast imagery**: high contrast between environment and host star.
- **Instrument**: Spectro-Polarimetric High-contrast Exoplanet REsearch (SPHERE) and its instrument InfraRed Dual Imaging and Spectrograph (IRDIS) installed on the Very Large Telescope (VLT).
- **Direct model**:

$$\mathrm{z}_{j,\ell} = T_{j,\ell} A \left(\frac{1}{2} + \cos^2(\theta - 2\alpha_\ell - \psi_j) \right) + \varepsilon_{j,\ell},$$

or

$$\mathrm{z}_{j,\ell} = \sum_{m=1}^{3} \nu_{j,\ell,m} \, T_{j,\ell} A \, S_m + \varepsilon_{j,\ell} + \varepsilon_{j,\ell},$$

DeepPDNet for high-contrast recovery

- **Analysis formulation:** $\boxed{\min_{\mathrm{x}} \frac{1}{2}\|\mathrm{Ax}-\mathrm{z}\|_2^2 + \|\mathrm{Lx}\|_1}$

- **Condat-Vũ iterations:**

$$\left|\begin{array}{ll} \mathrm{x}^{[k+1]} & = \mathrm{x}_k - \tau \mathrm{A}^*(\mathrm{Ax}^{[k]} - \mathrm{z}) - \tau \mathrm{L}^* \mathbf{u}^{[k]} \\ \mathbf{u}^{[k+1]} & = \mathrm{prox}_{\gamma\|\cdot\|_1^*}\left(\mathbf{u}^{[k]} + \gamma \mathrm{L}(2\mathrm{x}^{[k+1]} - \mathrm{x}^{[k]})\right) \end{array}\right.$$

- **Reformulation:**

$$\left|\begin{array}{l} \mathrm{x}^{[k+1]} = (\mathrm{Id} - \tau \mathrm{A}^*\mathrm{A})\mathrm{x}^{[k]} - \tau \mathrm{L}^* \mathbf{u}^{[k]} + \tau \mathrm{A}^* \mathrm{z} \\ \mathbf{u}^{[k+1]} = \mathrm{prox}_{\gamma\|\cdot\|_1^*}\left(\gamma \mathrm{L}(\mathrm{Id} - 2\tau \mathrm{A}^*\mathrm{A})\mathrm{x}^{[k]} + (\mathrm{Id} - 2\tau\gamma \mathrm{LL}^*)\mathbf{u}^{[k]} + 2\tau\gamma \mathrm{LA}^* \mathrm{z}\right). \end{array}\right.$$

- **Layer network:** [Jiu, Pustelnik, 2021]

$$\begin{bmatrix} \mathrm{x}^{[k+1]} \\ \mathbf{u}^{[k+1]} \end{bmatrix} = \underbrace{\boxed{\begin{matrix} \mathrm{Id} \\ \mathrm{prox}_{\gamma\|\cdot\|_1^*} \end{matrix}}}_{\eta^{[k]}} \left(\underbrace{\boxed{\begin{bmatrix} \mathrm{Id} - \tau \mathrm{A}^*\mathrm{A} & -\tau \mathrm{L}^* \\ \gamma \mathrm{L}(\mathrm{Id} - 2\tau \mathrm{A}^*\mathrm{A}) & \mathrm{Id} - 2\tau\gamma \mathrm{LL}^* \end{bmatrix}}}_{W^{[k]}} \begin{bmatrix} \mathrm{x}^{[k]} \\ \mathbf{u}^{[k]} \end{bmatrix} + \underbrace{\boxed{\begin{bmatrix} \tau \mathrm{A}^* \mathrm{z} \\ 2\tau\gamma \mathrm{LA}^* \mathrm{z} \end{bmatrix}}}_{b^{[k]}} \right)$$

DeepPDNet for high-contrast recovery

$$\boxed{f_{\Theta}(\mathrm{z}) = \eta^{[K]}\big(W^{[K]}\ldots\eta^{[1]}(W^{[1]}\mathrm{z}+b^{[1]})\ldots+b^{[K]}\big)}$$

☛ **Network with fixed layer:** $\Theta = \{\mathrm{L}, \tau, \gamma\}$

$$\begin{bmatrix}\mathrm{x}^{[k+1]}\\ \mathbf{u}^{[k+1]}\end{bmatrix} = \underbrace{\boxed{\begin{matrix}\mathrm{Id}\\ \mathrm{prox}_{\gamma\|\cdot\|_1^*}\end{matrix}}}_{\eta^{[k]}}\left(\underbrace{\boxed{\begin{bmatrix}\mathrm{Id}-\tau\mathrm{A}^*\mathrm{A} & -\tau\mathrm{L}^*\\ \gamma\mathrm{L}(\mathrm{Id}-2\tau\mathrm{A}^*\mathrm{A}) & \mathrm{Id}-2\tau\gamma\mathrm{L}\mathrm{L}^*\end{bmatrix}}}_{W^{[k]}}\begin{bmatrix}\mathrm{x}^{[k]}\\ \mathbf{u}^{[k]}\end{bmatrix} + \underbrace{\boxed{\begin{bmatrix}\tau\mathrm{A}^*\mathrm{z}\\ 2\tau\gamma\mathrm{L}\mathrm{A}^*\mathrm{z}\end{bmatrix}}}_{b^{[k]}}\right)$$

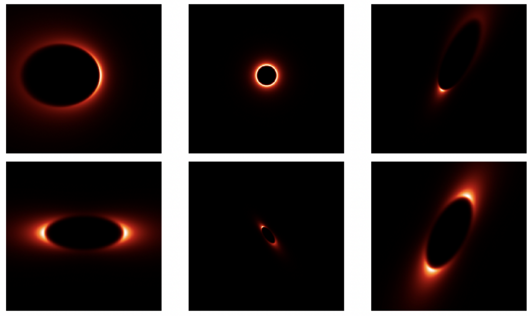
☛ **Network with variable layers:** $\Theta = \{\mathrm{L}_k, \tau_k, \gamma_k, \}_{1\leq k\leq K}$

$$\begin{bmatrix}\mathrm{x}^{[k+1]}\\ \mathbf{u}^{[k+1]}\end{bmatrix} = \underbrace{\boxed{\begin{matrix}\mathrm{Id}\\ \mathrm{prox}_{\gamma_k\|\cdot\|_1^*}\end{matrix}}}_{\eta^{[k]}}\left(\underbrace{\boxed{\begin{bmatrix}\mathrm{Id}-\tau_k\mathrm{A}^*\mathrm{A} & -\tau_k\mathrm{L}_k^*\\ \gamma_k\mathrm{L}_k(\mathrm{Id}-2\tau_k\mathrm{A}^*\mathrm{A}) & \mathrm{Id}-2\tau\gamma_k\mathrm{L}_k\mathrm{L}_k^*\end{bmatrix}}}_{W^{[k]}}\begin{bmatrix}\mathrm{x}^{[k]}\\ \mathbf{u}^{[k]}\end{bmatrix} + \underbrace{\boxed{\begin{bmatrix}\tau_k\mathrm{A}_k^*\mathrm{z}\\ 2\tau_k\gamma_k\mathrm{L}_k\mathrm{A}^*\mathrm{z}\end{bmatrix}}}_{b^{[k]}}\right)$$

$+$ specificities for the first and last layers.

Dataset

- **DDIT**: Debris DIsks Tools library produces synthetic images of $(_{\text{disk}}, , \theta)$.
- $_{\text{star}}$ has been obtained from real observational high-contrast coronagraphic data from the SPHERE.
- Different semi-major axis of the disk, inclination, eccentricity, and ratio between the star and disk intensity.



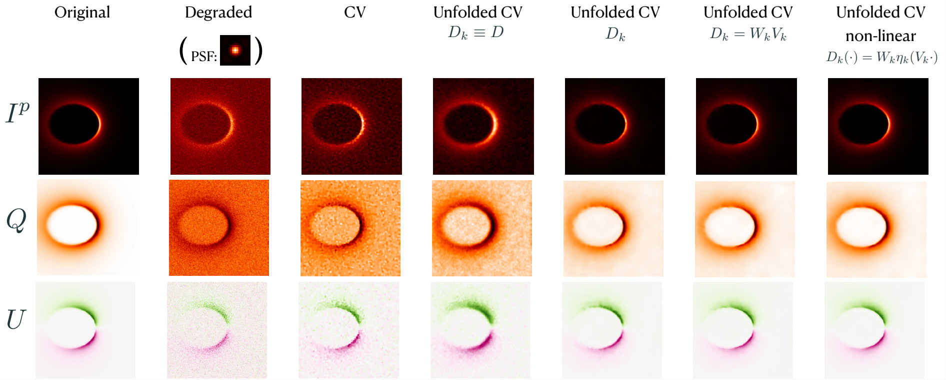
Dataset

- **DDIT**: Debris DIsks Tools library produces synthetic images of $(_{\text{disk}}, , \theta)$.
- $_{\text{star}}$ has been obtained from real observational high-contrast coronagraphic data from the SPHERE.
- Different semi-major axis of the disk, inclination, eccentricity, and ratio between the star and disk intensity.

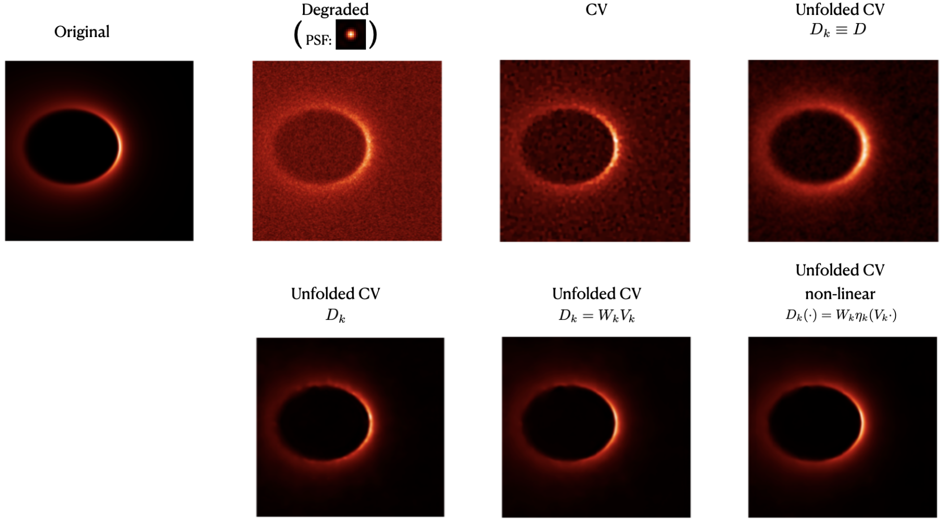
- **Synthetic dataset** – Prescribe blur and Gaussian noise with a standard deviation of 0.1.

- **More realistic dataset** – z obtained from RHAPSODIE forward model.

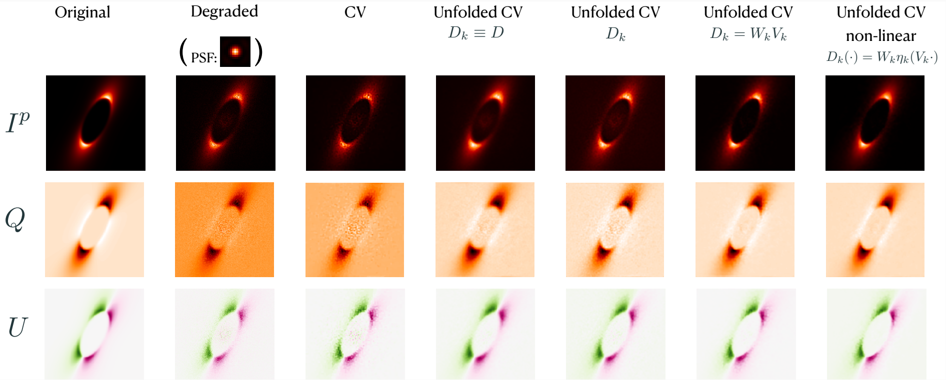
DeepPDNet for high-contrast recovery



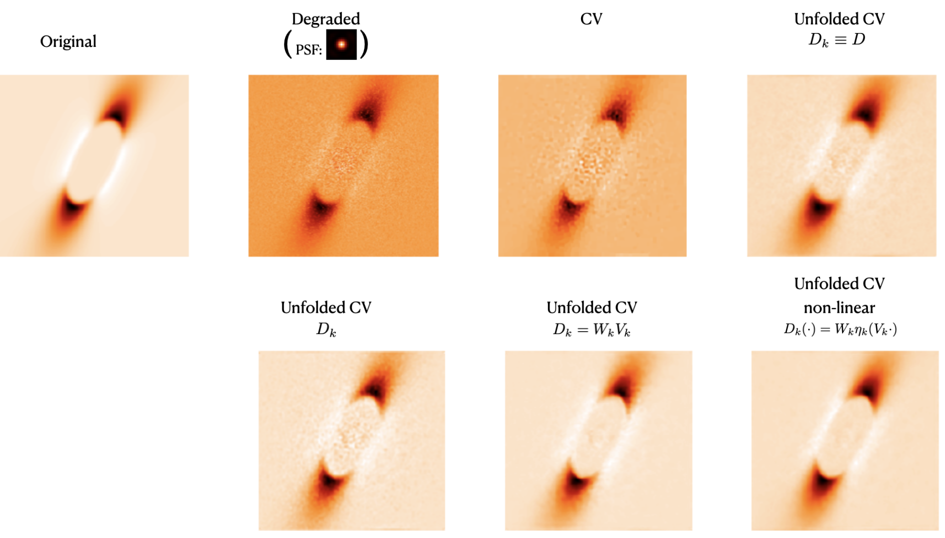
DeepPDNet for high-contrast recovery



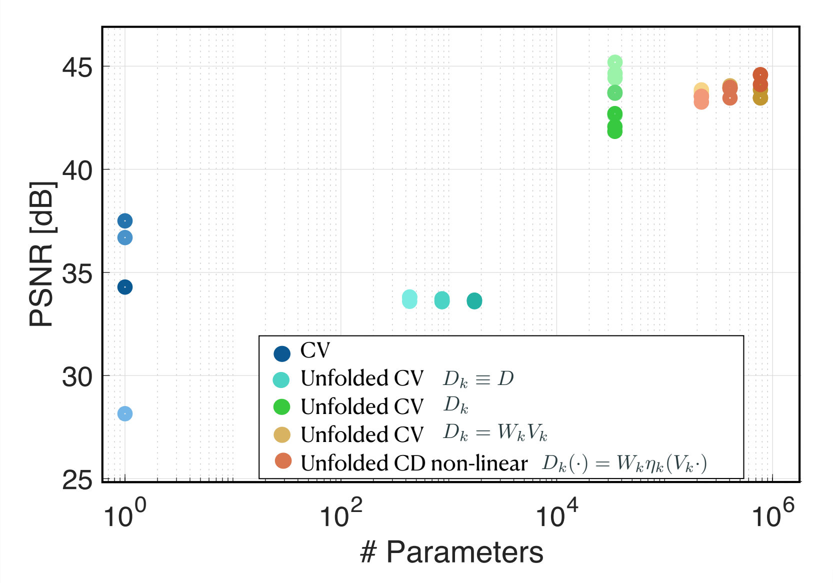
DeepPDNet for high-contrast recovery



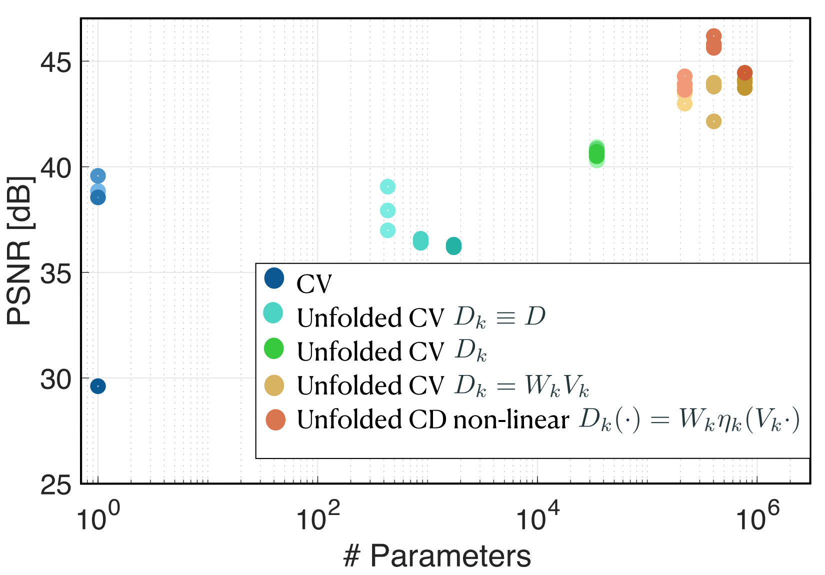
DeepPDNet for high-contrast recovery



DeepPDNet for high-contrast recovery



DeepPDNet for high-contrast recovery



References

- M. Jiu and N. Pustelnik, A deep primal-dual proximal network for image restoration, IEEE JSTSP Special Issue on Deep Learning for Image/Video Restoration and Compression, vol. 15, no. 2,pp. 190–203, Feb. 2021.
- M. Jiu, N. Pustelnik, Alternative Design of DeepPDNet in the Context of Image Restoration, IEEE Signal Processing Letters, vol. 29, pp. 932 - 936, 2022.
- H.T.V. Le, N. Pustelnik, M Foare, The faster proximal algorithm, the better unfolded deep learning architecture? The study case of image denoising, EUSIPCO, Belgrade, Serbie, 29 Aug - 2 Sept. 2022.
- H.T.V. Le, A. Repetti, N. Pustelnik, Unfolded proximal neural networks for robust image Gaussian denoising, IEEE TIP, 2024.
- E. Chappon, N. Pustelnik, J. Tachella, L. Denneulin, A. Ferarri, M. Langlois. PFrom linear to nonlinear unfolded Condat-Vũ algorithm for spectro-polarimetric hight-constrast image recovery, EUSIPCO, 2024.