An Analytical Theory for Inverse Problem CNN Solvers

Pierre Weiss, CNRS, Université de Toulouse
Joint work with Minh Hai Nguyen, Quoc Bao Do & Edouard Pauwels

Introduction

Inverse problems

$$y = A\bar{x} + e$$

- $\bar{x} \in \mathbb{R}^N$: signal to recover
- $A : \mathbb{R}^N \to \mathbb{R}^M$: linear operator
- $e \in \mathbb{R}^M$: noise drawn from $\mathcal{N}(0, \sigma^2 \mathrm{Id})$

Inverse problems

$$y = A\bar{x} + e$$

- $\bar{x} \in \mathbb{R}^N$: signal to recover
- $A : \mathbb{R}^N \to \mathbb{R}^M$: linear operator
- $e \in \mathbb{R}^M$: noise drawn from $\mathcal{N}(0, \sigma^2 \mathrm{Id})$

Lack of identifiability

Information lost through A

Need for a prior probability distribution $p_{\mathbf{x}} : \mathbb{R}^N \to \mathbb{R}$

Supervised learning ≈ approximating $p_{\mathbf{x}}$

MAP VS MMSE

$$\hat{x}_{\mathrm{MAP}} \overset{\text{def.}}{=} \operatorname*{argmax}_{x} p_{\mathbf{x}|\mathbf{y}}(x|y)$$

$$= \operatorname*{argmin}_{x \in \mathbb{R}^N} \frac{\|Ax - y\|^2}{2\sigma^2} - \log p_{\mathbf{x}}(x)$$

- Optimization
- Nice looking
- Local minimizers/Unstable

MAP VS MMSE

$$\hat{x}_{\mathrm{MAP}} \overset{\text{def.}}{=} \underset{x}{\operatorname{argmax}}\, p_{\boldsymbol{x}|\boldsymbol{y}}(x|y)$$

$$= \underset{x\in\mathbb{R}^N}{\operatorname{argmin}} \frac{\|Ax-y\|^2}{2\sigma^2} - \log p_{\boldsymbol{x}}(x)$$

- Optimization
- Nice looking
- Local minimizers/Unstable

$$\hat{x}_{\mathrm{MMSE}} \overset{\text{def.}}{=} \underset{x}{\operatorname{argmin}}\, \mathbb{E}[\|x-\boldsymbol{x}\|_2^2|\boldsymbol{y}=y]$$

$$= \int x \cdot d_{p_{\boldsymbol{x}|\boldsymbol{y}=y}(x)}$$

- Integration
- More stability
- Blurry where dubious

MAP VS MMSE

$$\hat{x}_{\mathrm{MAP}} \overset{\text{def.}}{=} \operatorname*{argmax}_{x} p_{\boldsymbol{x}|\boldsymbol{y}}(x|y)$$

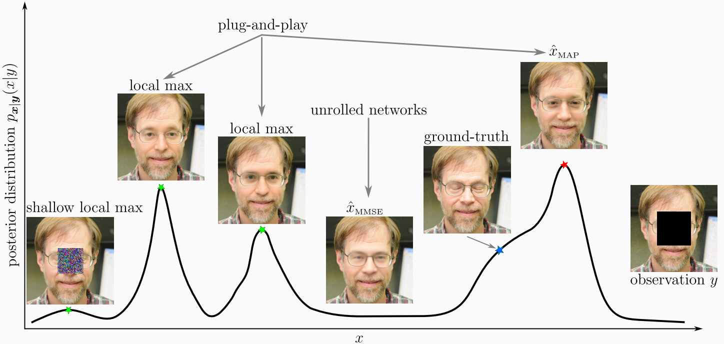
$$= \operatorname*{argmin}_{x \in \mathbb{R}^N} \frac{\|Ax - y\|^2}{2\sigma^2} - \log p_{\boldsymbol{x}}(x)$$

- Optimization
- Nice looking
- Local minimizers/Unstable

$$\hat{x}_{\mathrm{MMSE}} \overset{\text{def.}}{=} \operatorname*{argmin}_{x} \mathbb{E}[\|x - \boldsymbol{x}\|_2^2 | \boldsymbol{y} = y]$$

$$= \int x \cdot d_{p_{\boldsymbol{x}|\boldsymbol{y}=y}(x)}$$

- Integration
- More stability
- Blurry where dubious



MAP VS MMSE

$$\hat{x}_{\mathrm{MAP}} \overset{\text{def.}}{=} \operatorname*{argmax}_{x} p_{\boldsymbol{x}|\boldsymbol{y}}(x|y)$$

- Intense use $\approx$ 100 years
- e.g. Proximal methods (convex)
- "Plug & Play" methods
- Sampling $p_{\boldsymbol{x}|\boldsymbol{y}}$ via Langevin dynamics / diffusion
- Heavy at training (learn $p_{\boldsymbol{x}}$)
- Heavy at inference

MAP VS MMSE

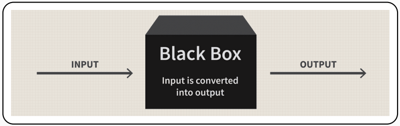
$$\hat{x}_{\mathrm{MAP}} \overset{\text{def.}}{=} \underset{x}{\operatorname{argmax}}\, p_{\mathbf{x}|\mathbf{y}}(x|y)$$

- Intense use $\approx$ 100 years
- e.g. Proximal methods (convex)
- "Plug & Play" methods
- Sampling $p_{\mathbf{x}|\mathbf{y}}$ via Langevin dynamics / diffusion
- Heavy at training (learn $p_{\mathbf{x}}$)
- Heavy at inference

$$\hat{x}_{\mathrm{MMSE}} \overset{\text{def.}}{=} \mathbb{E}[\mathbf{x}|\mathbf{y} = y]$$
$$= \int x \cdot d_{p_{\mathbf{x}|\mathbf{y}=y}(x)}$$

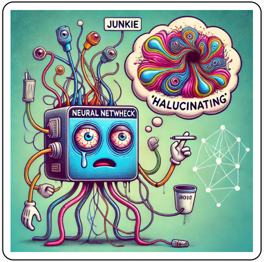
- -2015: Gibbs = too heavy
- 2015-: Supervised learning
  - AutoMAP
  - Unrolled
  - RAM...
- Heavy training
- Fast at inference

My understanding: for non generative science MMSE currently better

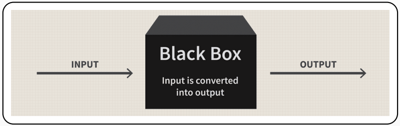


Cons

- Neural nets are black boxes
- Neural nets hallucinate [1]

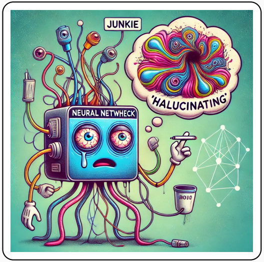


[1] Gottschling et al. The troublesome kernel: On hallucinations, no free lunches, and the accuracy-stability tradeoff in inverse problems, SIAM Review 2025
[2] Genzel et al. Solving inverse problems with deep neural networks–robustness included?, IEEE PAMI 2022



Cons

- Neural nets are black boxes
- Neural nets hallucinate [1]



Pros

- Well trained neural nets are empirically stable
- Well trained neural nets are more stable than TV [2]
- Current state-of-the-art
- FDA approval in 1 year compared to 10 years for ℓ^1

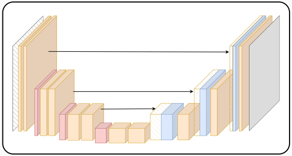
[1] Gottschling et al. The troublesome kernel: On hallucinations, no free lunches, and the accuracy-stability tradeoff in inverse problems, SIAM Review 2025
[2] Genzel et al. Solving inverse problems with deep neural networks–robustness included?, IEEE PAMI 2022

Supervised learning for inverse problems

- $\boldsymbol{e} \sim \mathcal{N}(0, \sigma^2 \mathrm{Id})$
- $\boldsymbol{y} = A\boldsymbol{x} + \boldsymbol{e}$: noisy image.

Supervised learning for inverse problems

- $\boldsymbol{e} \sim \mathcal{N}(0, \sigma^2 \mathrm{Id})$
- $\boldsymbol{y} = A\boldsymbol{x} + \boldsymbol{e}$: noisy image.

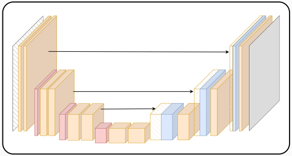


Prerequisities

- Neural network $N(y, A, \theta)$.
- A database $\mathcal{D} = (x_1, \ldots, x_D)$
- Empirical measure: $p_{\mathcal{D}} = \frac{1}{D} \sum_{x \in \mathcal{D}} \delta_x$
- Synthesize $\boldsymbol{y} = A\boldsymbol{x} + \boldsymbol{e}$, $\boldsymbol{x} \sim p_{\mathcal{D}}$

Supervised learning for inverse problems

- $\boldsymbol{e} \sim \mathcal{N}(0, \sigma^2 \mathrm{Id})$
- $\boldsymbol{y} = A\boldsymbol{x} + \boldsymbol{e}$: noisy image.



Prerequisities

- Neural network $N(y, A, \theta)$.
- A database $\mathcal{D} = (x_1, \ldots, x_D)$
- Empirical measure: $p_{\mathcal{D}} = \frac{1}{D} \sum_{x \in \mathcal{D}} \delta_x$
- Synthesize $\boldsymbol{y} = A\boldsymbol{x} + \boldsymbol{e}$, $\boldsymbol{x} \sim p_{\mathcal{D}}$

Training $\equiv$ Stochastic gradient

- $\inf_{\theta} \mathbb{E}_{\boldsymbol{x},\boldsymbol{y}}[\|N(\boldsymbol{y}, A, \theta) - \boldsymbol{x}\|_2^2]$
- Note: infinite number of noise realizations

Output

- $N(\cdot, A, \theta^\star)$: a trained network
- Can be used for any input y

Minimum Mean Square Estimation (MMSE)

$$\hat{x}_{MMSE} \stackrel{\text{def.}}{=} \underset{\phi:\mathbb{R}^M \to \mathbb{R}^N}{\operatorname{argmin}} \mathbb{E}_{\boldsymbol{x},\boldsymbol{y}}(\|\phi(\boldsymbol{y}) - \boldsymbol{x}\|_2^2)$$

Minimum Mean Square Estimation (MMSE)

$$\hat{x}_{MMSE} \overset{\text{def.}}{=} \underset{\phi:\mathbb{R}^M \to \mathbb{R}^N}{\operatorname{argmin}} \mathbb{E}_{\boldsymbol{x},\boldsymbol{y}}(\|\phi(\boldsymbol{y}) - \boldsymbol{x}\|_2^2)$$

Claim (informal)

If D large enough, $N(\cdot, A, \theta)$ expressive + generalizes, good training.

$$N(y, A, \theta^\star) \approx \hat{x}_{MMSE}(y)$$

Minimum Mean Square Estimation (MMSE)

$$\hat{x}_{MMSE} \overset{\text{def.}}{=} \underset{\phi:\mathbb{R}^M \to \mathbb{R}^N}{\operatorname{argmin}} \mathbb{E}_{\boldsymbol{x},\boldsymbol{y}}(\|\phi(\boldsymbol{y}) - \boldsymbol{x}\|_2^2)$$

Claim (informal)

If D large enough, $N(\cdot, A, \theta)$ expressive + generalizes, good training.

$$N(y, A, \theta^\star) \approx \hat{x}_{MMSE}(y)$$

Proof.

$$N(\cdot, A, \theta^\star) \approx \underset{\phi=N(\cdot,A,\theta)}{\operatorname{argmin}} \mathbb{E}\left[\frac{1}{D}\sum_{d=1}^{D} \|\phi(\boldsymbol{y}_d) - x_d\|_2^2\right] \qquad \text{Good optimization}$$

$$\approx \underset{\phi \text{ measurable}}{\operatorname{argmin}} \mathbb{E}\left[\frac{1}{D}\sum_{d=1}^{D} \|\phi(\boldsymbol{y}_d) - x_d\|_2^2\right] \qquad \text{Expressivity}$$

$$\approx \underset{\phi \text{ measurable}}{\operatorname{argmin}} \mathbb{E}_{\boldsymbol{x},\boldsymbol{y}}(\|\phi(\boldsymbol{y}) - \boldsymbol{x}\|_2^2) \qquad p_{\mathcal{D}} \rightharpoonup p_{\boldsymbol{x}} + \text{Generalization}$$

$$\overset{\text{def.}}{=} \hat{x}_{MMSE}!$$

Minimum Mean Square Estimation (MMSE)

$$\hat{x}_{MMSE} \overset{\text{def.}}{=} \underset{\phi:\mathbb{R}^M \to \mathbb{R}^N}{\operatorname{argmin}} \mathbb{E}_{\boldsymbol{x},\boldsymbol{y}}(\|\phi(\boldsymbol{y}) - \boldsymbol{x}\|_2^2)$$

Claim (informal)

If D large enough, $N(\cdot, A, \theta)$ expressive + generalizes, good training.

$$N(y, A, \theta^\star) \approx \hat{x}_{MMSE}(y)$$

Proof.

$$N(\cdot, A, \theta^\star) \approx \underset{\phi = N(\cdot, A, \theta)}{\operatorname{argmin}} \mathbb{E}\left[\frac{1}{D}\sum_{d=1}^{D} \|\phi(\boldsymbol{y}_d) - x_d\|_2^2\right] \qquad \text{Good optimization}$$

$$\approx \underset{\phi \text{ measurable}}{\operatorname{argmin}} \mathbb{E}\left[\frac{1}{D}\sum_{d=1}^{D} \|\phi(\boldsymbol{y}_d) - x_d\|_2^2\right] \qquad \text{Expressivity}$$

$$\approx \underset{\phi \text{ measurable}}{\operatorname{argmin}} \mathbb{E}_{\boldsymbol{x},\boldsymbol{y}}(\|\phi(\boldsymbol{y}) - \boldsymbol{x}\|_2^2) \qquad p_{\mathcal{D}} \rightharpoonup p_{\boldsymbol{x}} + \text{Generalization}$$

$$\overset{\text{def.}}{=} \hat{x}_{MMSE}!$$

Not observed in practice! What happens with less expressivity and D finite?

Preliminaries

Our neural networks

$$\Phi_\theta : \mathbb{R}^N \to \mathbb{R}^N \quad \text{with} \quad \Phi_\theta(x) = \rho \circ A_K \dots \rho \circ A_2 \circ \rho \circ A_1 \circ x$$

- ρ: component-wise activation function (ReLU, sigmoid, ...)
- A_k: affine operators.
- θ: network weights $(A_1, \dots, A_K)$
- K: number of layers

Our neural networks

$$\Phi_\theta : \mathbb{R}^N \to \mathbb{R}^N \quad \text{with} \quad \Phi_\theta(x) = \rho \circ A_K \ldots \rho \circ A_2 \circ \rho \circ A_1 \circ x$$

- ρ: component-wise activation function (ReLU, sigmoid, ...)
- A_k: affine operators.
- θ: network weights $(A_1, \ldots, A_K)$
- K: number of layers

Covers various architectures

- A_k affine: Multi-Layer Perceptron (MLP)
- A_k convolution: Convolutional Neural Network (CNN)
- A_k local convolution: local CNN (most common)

Our neural networks

Adaptation to inverse problems (solid baseline)

$A : \mathbb{R}^N \to \mathbb{R}^M$, so take an "inverse" $B : \mathbb{R}^M \to \mathbb{R}^N$

- **Physics-agnostic**[3] $M = N$, $B = \mathrm{Id}$

$$\hat{x}(y) = \Phi_\theta(y)$$

- **Physics-aware**[4] $B = A^T$, $B = A^+$, Tikhonov-regularized inverse..

$$\hat{x}(y) = \Phi_\theta(By)$$

[3] Zhu et al. Image reconstruction by domain-transform manifold learning, Nature 2018
[4] Mc Cann et al. Deep CNN for inverse problems in imaging, IEEE IP 2017

Set $\mu \in \mathbb{R}^N$ and $\Sigma \in \mathbb{R}^{N \times N}$ symmetric, positive semi-definite matrix of rank r.

Let Σ^+ be its pseudo-inverse and $|\Sigma|_+$ denote the pseudo-determinant.

Definition (Generalized Gaussian distribution)

The Gaussian distribution is:

$$\mathcal{N}(z; \mu, \Sigma) \stackrel{\text{def.}}{=} \begin{cases} \frac{\exp\left(-\frac{1}{2}(z-\mu)^T \Sigma^+ (z-\mu)\right)}{\sqrt{(2\pi)^r |\Sigma|_+}} & \text{if } z \in \mu + \mathrm{Im}\Sigma \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

defined w.r.t to the Hausdorff measure on $\mu + \mathrm{Im}\Sigma$.

Multi Layer Perceptrons (MLPs)

Analytical solutions for MLPs

Multi Layer Perceptron (MLP)

Let $\mathcal{R}_\Phi = \{\Phi_\theta, \theta \in \Theta\}$ denote the range of a neural network.

For an MLP, universal approximation theorem states:

$$\mathcal{R}_\Phi \approx \mathcal{M} \overset{\text{def.}}{=} \{\phi \ \text{ continuous from } \mathbb{R}^N \to \mathbb{R}^N\}$$

Analytical solutions for MLPs

Multi Layer Perceptron (MLP)

Let $\mathcal{R}_\Phi = \{\Phi_\theta, \theta \in \Theta\}$ denote the range of a neural network.

For an MLP, universal approximation theorem states:

$$\mathcal{R}_\Phi \approx \mathcal{M} \overset{\text{def.}}{=} \{\phi \;\text{ continuous from } \mathbb{R}^N \to \mathbb{R}^N\}$$

Theorem (MLP & MMSE estimator)

Set $\hat{x}_{MMSE}(y) = \phi^\star(By)$ with $\phi^\star$ the solution to

$$\phi^\star \overset{\text{def.}}{=} \operatorname*{argmin}_{\phi \in \mathcal{M}} \frac{1}{D} \sum_{d=1}^{D} \mathbb{E}\left(\|\phi(B\boldsymbol{y}_d) - x_d\|^2\right).$$

$$\hat{x}_{MMSE}(y) = \sum_{d=1}^{D} x_d \cdot w_d \quad \text{where} \quad w_d \overset{\text{def.}}{=} \mathcal{N}\left(By; BAx_d, \sigma^2 BB^T\right) / Z(y)$$

with $Z(y) \overset{\text{def.}}{=} \sum_{d'=1}^{D} \mathcal{N}\left(By; BAx_d, \sigma^2 BB^T\right) \propto p_y(y)$.

Some Remarks for MLPs

$$w_d \propto \mathcal{N}\left(By; BAx_d, \sigma^2 BB^T\right) \propto \exp\left(-\frac{\|By - BAx_d\|^2_{BB^T}}{2\sigma^2}\right)$$

- Weights $\propto \exp\left(-\frac{1}{2\sigma^2}\|\Pi_{\mathrm{Im}(B)}(y - Ax)\|^2\right)$
- $B = A^+$ or A^T is most natural: $\|A(x - \bar{x}) + \Pi_{\ker(A)^\perp} e\|$

[5] Vastola Generalization through variance: how noise shapes inductive biases in diffusion models , ICLR 2025

Analytical solutions for MLPs

Some Remarks for MLPs

$$w_d \propto \mathcal{N}\left(By; BAx_d, \sigma^2 BB^T\right) \propto \exp\left(-\frac{\|By - BAx_d\|^2_{BB^T}}{2\sigma^2}\right)$$

- Weights $\propto \exp\left(-\frac{1}{2\sigma^2}\|\Pi_{\mathrm{Im}(B)}(y - Ax)\|^2\right)$
- $B = A^+$ or A^T is most natural: $\|A(x - \bar{x}) + \Pi_{\ker(A)^\perp} e\|$
- $\lim_{\sigma \to 0^+} \hat{x}_{MMSE}(y)$ is $x \in \mathcal{D}$ with smallest $\|\Pi_{\mathrm{Im}(B)}(y - Ax)\|$.
- $\hat{x}_{MMSE}(y) \in \mathrm{Conv}(\mathcal{D})$
- Moderate $\sigma \Rightarrow$ analytical formula copy-pastes dataset elements

[5] Vastola Generalization through variance: how noise shapes inductive biases in diffusion models , ICLR 2025

Analytical solutions for MLPs

Some Remarks for MLPs

$$w_d \propto \mathcal{N}\left(By; BAx_d, \sigma^2 BB^T\right) \propto \exp\left(-\frac{\|By - BAx_d\|^2_{BB^T}}{2\sigma^2}\right)$$

- Weights $\propto \exp\left(-\frac{1}{2\sigma^2}\|\Pi_{\mathrm{Im}(B)}(y - Ax)\|^2\right)$
- $B = A^+$ or A^T is most natural: $\|A(x - \bar{x}) + \Pi_{\ker(A)^\perp} e\|$
- $\lim_{\sigma \to 0^+} \hat{x}_{MMSE}(y)$ is $x \in \mathcal{D}$ with smallest $\|\Pi_{\mathrm{Im}(B)}(y - Ax)\|$.
- $\hat{x}_{MMSE}(y) \in \mathrm{Conv}(\mathcal{D})$
- Moderate $\sigma \Rightarrow$ analytical formula copy-pastes dataset elements
- The MMSE (MLP) memorizes the dataset!

[5] Vastola Generalization through variance: how noise shapes inductive biases in diffusion models , ICLR 2025

Analytical solutions for MLPs

Some Remarks for MLPs

$$w_d \propto \mathcal{N}\left(By; BAx_d, \sigma^2 BB^T\right) \propto \exp\left(-\frac{\|By - BAx_d\|_{BB^T}^2}{2\sigma^2}\right)$$

- Weights $\propto \exp\left(-\frac{1}{2\sigma^2}\|\Pi_{\mathrm{Im}(B)}(y - Ax)\|^2\right)$
- $B = A^+$ or A^T is most natural: $\|A(x - \bar{x}) + \Pi_{\ker(A)^\perp} e\|$
- $\lim_{\sigma \to 0^+} \hat{x}_{MMSE}(y)$ is $x \in \mathcal{D}$ with smallest $\|\Pi_{\mathrm{Im}(B)}(y - Ax)\|$.
- $\hat{x}_{MMSE}(y) \in \mathrm{Conv}(\mathcal{D})$
- Moderate $\sigma \Rightarrow$ analytical formula copy-pastes dataset elements
- The MMSE (MLP) memorizes the dataset!

Is this formula valid?

- Used heavily in influential papers [5]...
- We could not reproduce this behavior with neural networks

[5] Vastola Generalization through variance: how noise shapes inductive biases in diffusion models , ICLR 2025

Deep Convolutional Neural Networks (CNNs)

MMSE & CNN models

For deep CNNs, we have

$$\mathcal{R}_\Phi \approx \mathcal{M}_\mathcal{T} \overset{\text{def.}}{=} \left\{ \phi \in \mathcal{C}(\mathbb{R}^N, \mathbb{R}^N) \text{ translation-equivariant} \right\}$$

That is $\phi(T_g x) = T_g \phi(x), \quad \forall T_g$ circular translation

MMSE & CNN models

For deep CNNs, we have

$$\mathcal{R}_\Phi \approx \mathcal{M}_\mathcal{T} \overset{\text{def.}}{=} \left\{\phi \in \mathcal{C}(\mathbb{R}^N, \mathbb{R}^N) \text{ translation-equivariant}\right\}$$

That is $\phi(T_g x) = T_g \phi(x), \quad \forall T_g$ circular translation

Theorem

Set $\hat{x}_\mathcal{T}(y) = \phi^\star(By)$ with $\phi^\star$ the solution to

$$\phi^\star \overset{\text{def.}}{=} \underset{\phi \in \mathcal{M}_\mathcal{T}}{\operatorname{argmin}} \frac{1}{D} \sum_{d=1}^{D} \mathbb{E}\left(\|\phi(B\boldsymbol{y}_d) - x_d\|^2\right).$$

$$\hat{x}_\mathcal{T}(y) = \sum_{x \in \mathcal{D}, g \in \mathcal{T}} T_g x \cdot w_g(x|y) \quad \text{with} \quad w_g(x|y) = \frac{\mathcal{N}\left(T_g^{-1} By; BAx, \sigma^2 BB^T\right)}{Z(y)}$$

We average on the group-augmented dataset $\mathcal{T}\mathcal{D}$!

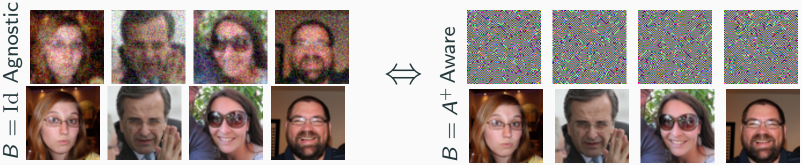
Choosing B = non trivial

Data augmentation $\neq$ architecture equivariance!

Equivalence iff A and B are invertible circular convolutions

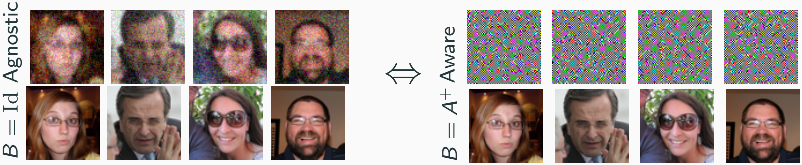
Deconvolution

Any invertible convolution B yields same result



Deconvolution

Any invertible convolution B yields same result



Inpainting

$$\hat{x}_{\mathcal{T}}(y) \approx \underset{x \in \mathcal{D}}{\operatorname{argmin}} \|A(x - y)\| \quad \text{for } B = A^{+}, \sigma \approx 0$$

Analytical formula VS CNN

We do not experience a good match between the formula and a trained CNN!

Analytical formula VS CNN

We do not experience a good match between the formula and a trained CNN!

Fitting vs generalization

The network is trained with a density:

$$p_y = \frac{1}{D} \sum_{x \in \mathcal{D}} \mathcal{N}(Ax, \sigma^2 \mathrm{Id}_M) = G_\sigma \star p_{A\mathcal{D}}$$

For small σ, high-dimensional x, $G_\sigma \star p_{A\mathcal{D}}$ is a poor approximation of $p_{Ax} \star G_\sigma$.

Formula for y far from $A\mathcal{D}$ does not fit.

Analytical formula VS CNN

We do not experience a good match between the formula and a trained CNN!

Fitting vs generalization

The network is trained with a density:

$$p_y = \frac{1}{D} \sum_{x \in \mathcal{D}} \mathcal{N}(Ax, \sigma^2 \mathrm{Id}_M) = G_\sigma \star p_{A\mathcal{D}}$$

For small σ, high-dimensional x, $G_\sigma \star p_{A\mathcal{D}}$ is a poor approximation of $p_{A\mathbf{x}} \star G_\sigma$.

Formula for y far from $A\mathcal{D}$ does not fit.

We should study the generalization regime, but this is not the scope here.

Convolutional Neural Networks (CNNs)

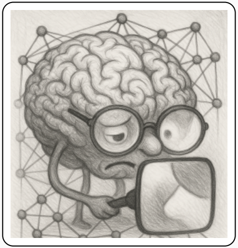
Locality & equivariance

For regular CNNs, Φ_θ has a finite receptive field ω, with $|\omega| = P$.

$$\mathcal{R}_\Phi \approx \mathcal{M}_{loc,\mathcal{T}} \overset{\text{def.}}{=} \left\{\phi \circ B : \phi_n = f \circ \Pi_n \,,\; f : \mathbb{R}^P \to \mathbb{R} \text{ is continuous}\right\}$$

where $\Pi_n x \overset{\text{def.}}{=} x[\omega_n]$ is a patch-extractor of $\omega_n = \omega \text{ shifted by } n$.

In words: value of pixel n only depends on $By[\omega_n]$.



[6] Formula gets awful otherwise with stratification

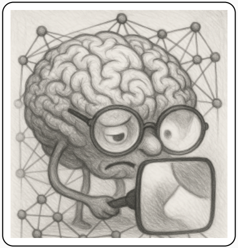
Locality & equivariance

For regular CNNs, Φ_θ has a finite receptive field ω, with $|\omega| = P$.

$$\mathcal{R}_\Phi \approx \mathcal{M}_{loc,\mathcal{T}} \stackrel{\text{def.}}{=} \left\{ \phi \circ B : \phi_n = f \circ \Pi_n \,,\; f : \mathbb{R}^P \to \mathbb{R} \text{ is continuous} \right\}$$

where $\Pi_n x \stackrel{\text{def.}}{=} x[\omega_n]$ is a patch-extractor of $\omega_n = \omega$ shifted by n.

In words: value of pixel n only depends on $By[\omega_n]$.



Theorem

Suppose $Q_n = \Pi_n B \in \mathbb{R}^{P\times M}$ have constant rank $r > 0$ [6].

At any pixel $n' \in [\![1, N]\!]$:

$$\hat{x}_{n'}(y) = \sum_{x\in\mathcal{D}} \sum_{n=1}^{N} x_n \cdot w_{n',n}(x|y)$$

$$w_{n',n}(x|y) = \frac{\mathcal{N}\left((By)[\omega_{n'}]; (BAx)[\omega_n], \sigma^2 Q_n Q_n^T\right)}{Z_{n'}(y)}.$$

[6] Formula gets awful otherwise with stratification

The case of denoising ($A = B = \mathrm{Id}$)

The weights become [7]

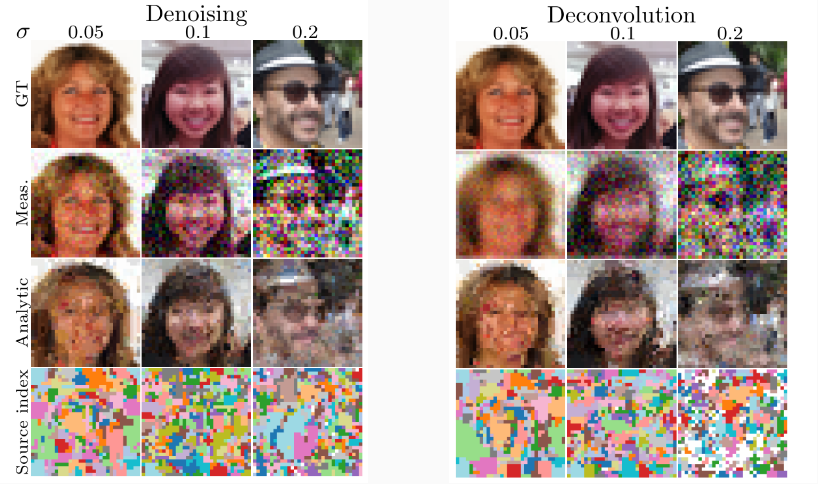
$$w_{n',n}(x|y) = \exp\left(-\left\|y[\omega_{n'}] - x[\omega_n]\right\|^2 / (2\sigma^2)\right) / Z_{n'}(y),$$

- $\sigma \to \infty$: average of all pixels in the dataset.
- $\sigma \to 0$: central pixel of dataset patch closest to $y[\omega_{n'}]$.
- Denoised images are patchworks of images in training database!

CNNs are cubist artists! (Here, Picasso)

[7] Ganguli & Kamb An analytic theory of creativity in CNN diffusion models , ICML 2025

CNN make patchworks of training images



Pixels corresponding to more than 50% of the mass

On the role of the pre-inverse B

Let $Q_n = \Pi_n B \in \mathbb{R}^{P \times N}$.

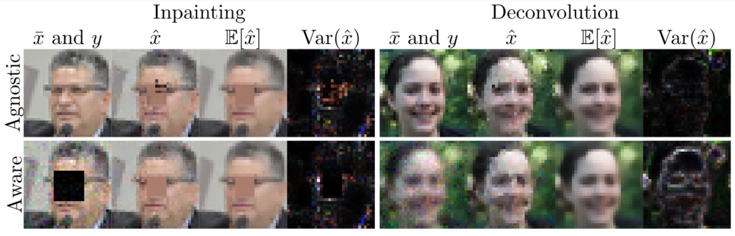
$$\Delta_{n',n}(\bar{x}, x) \stackrel{\text{def.}}{=} (BA\bar{x})[\omega_{n'}] - (BAx)[\omega_n]$$

the weights write:

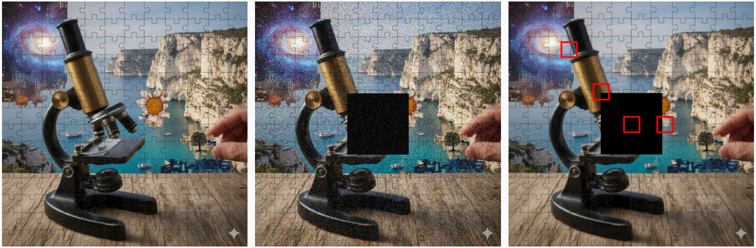
$$w_{n',n}(x|y) \propto \exp\left(-\frac{\eta^2}{2\sigma^2}\right) \quad \text{with} \quad \eta \stackrel{\text{def.}}{=} \left\| Q_n^+ \Delta_{n',n}(\bar{x}, x) + Q_n^+ Q_{n'} e \right\| .$$

The action of B:

- $Q_n^+ \Delta_{n',n}(\bar{x}, x)$: discriminate similar/dissimilar patches
- $Q_n^+ Q_{n'} e$: noise amplification/reduction

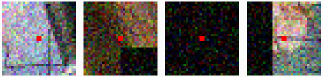


Locality & Inpainting

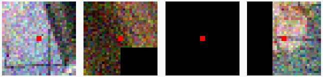


The inpainting problem $Ax = 1_{\Omega^c} \odot x$

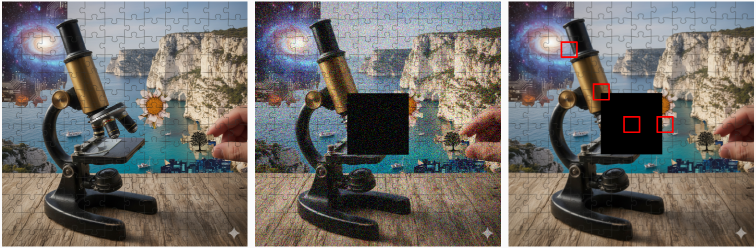
$B = \mathrm{Id}$



$B = A^+$

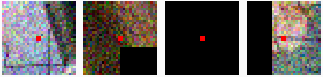


Locality & Inpainting



The inpainting problem $Ax = 1_{\Omega^c} \odot x$

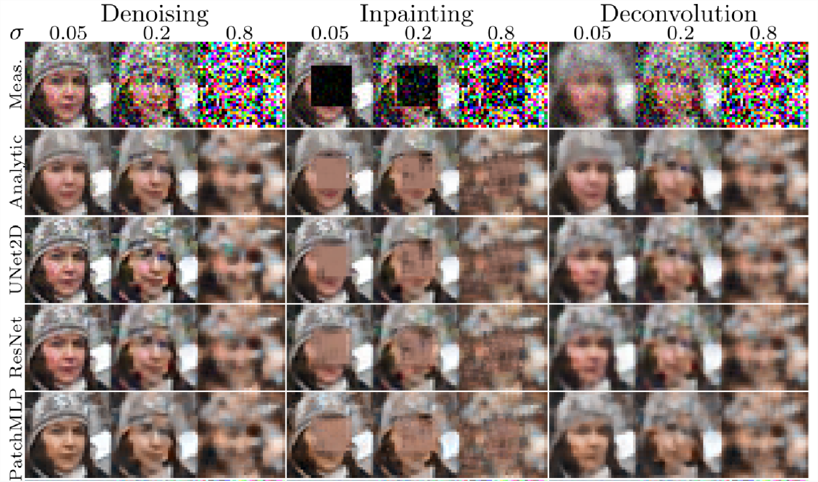
$B = A^+$



With $B = A^+$, at pixel n only pixels n' s.t. $\omega_{n'} \cap \text{mask} = \omega_n \cap \text{mask}$ are used

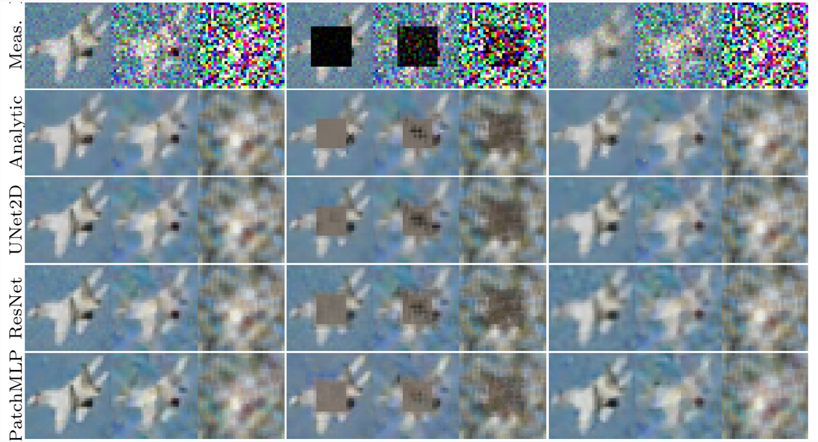
Theory VS practice

A good match!



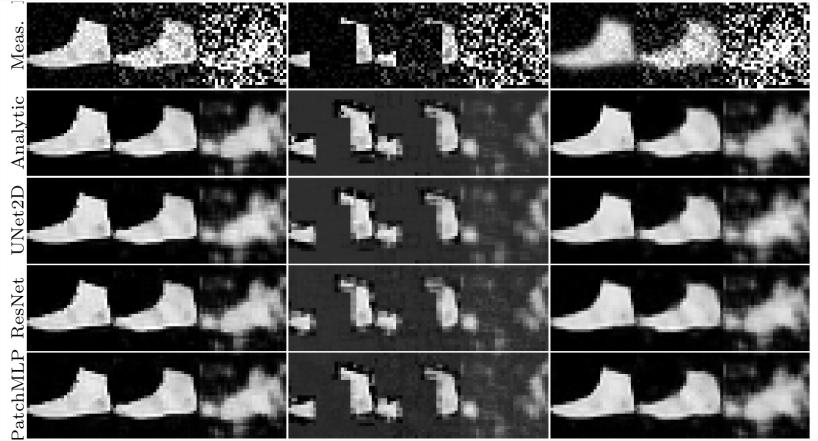
CNN vs analytical for $|\omega| = 5 \times 5$ and $N = 32 \times 32 \times 3$.

A good match!



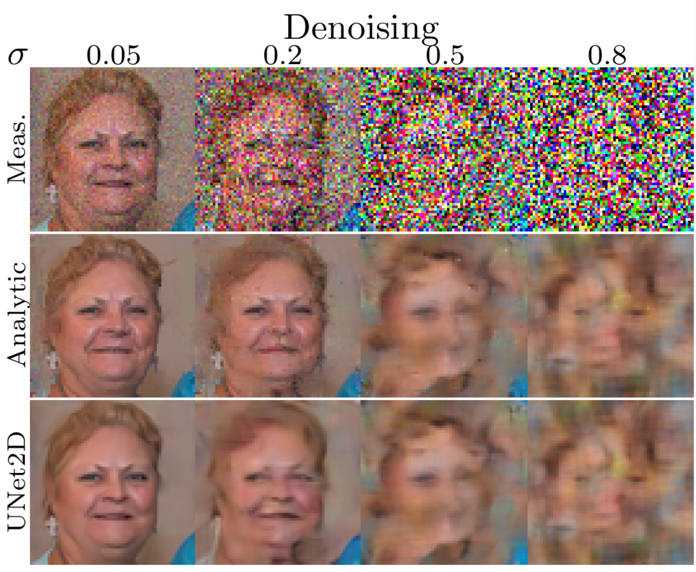
CNN vs analytical for $|\omega| = 5 \times 5$ and $N = 32 \times 32 \times 3$.

A good match!



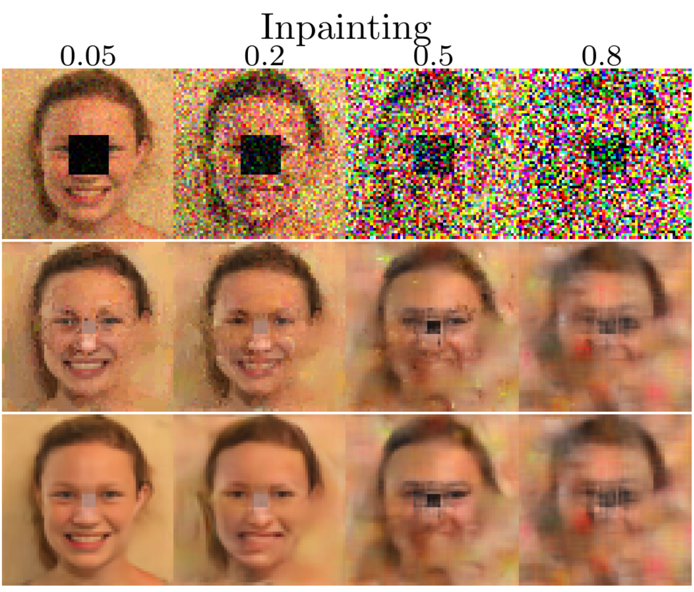
CNN vs analytical for $|\omega| = 5 \times 5$ and $N = 32 \times 32 \times 3$.

A good match!



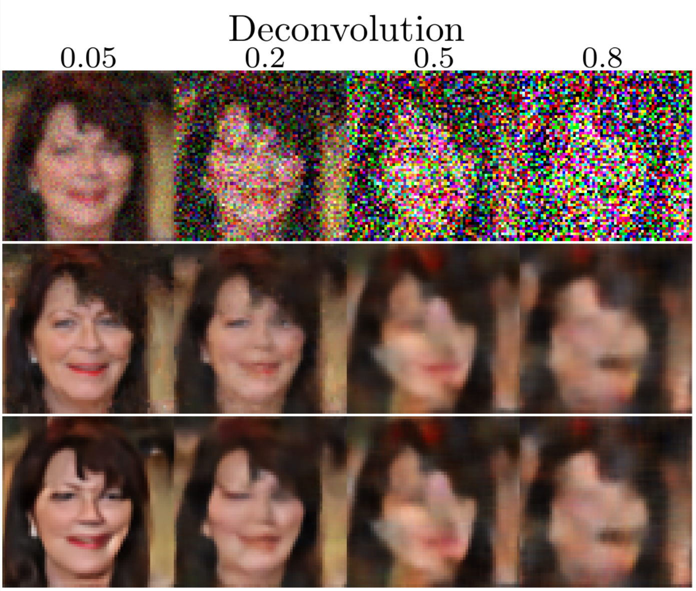
CNN vs analytical for $|\omega| = 11 \times 11$ and $N = 64 \times 64 \times 3$.

A good match!

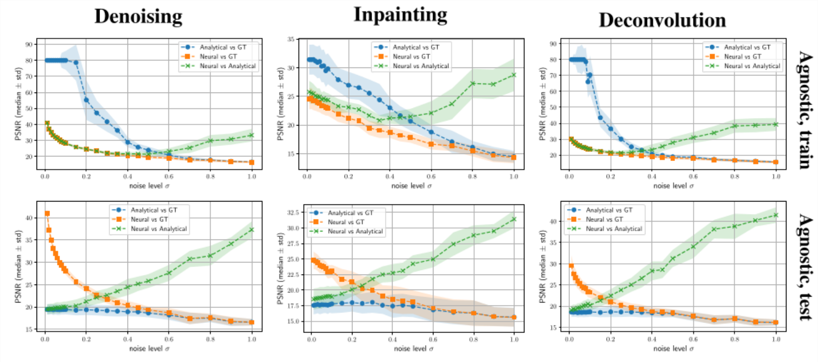


CNN vs analytical for $|\omega| = 11 \times 11$ and $N = 64 \times 64 \times 3$.

A good match!



CNN vs analytical for $|\omega| = 11 \times 11$ and $N = 64 \times 64 \times 3$.

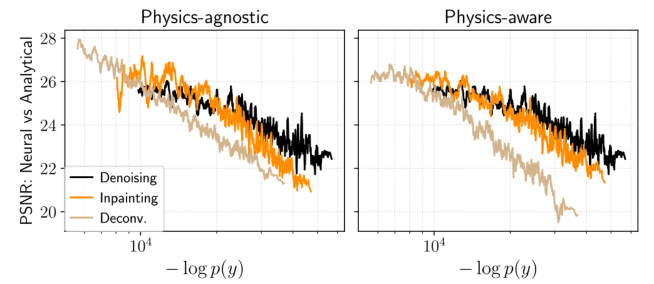


Theory vs practice

Experimental comparisons

- $< 1\%$ relative error, irrespective of noise level, inverse problem
- Closer match for large noise levels
- Better generalization capability for CNNs than analytical formula

The limit of the theory

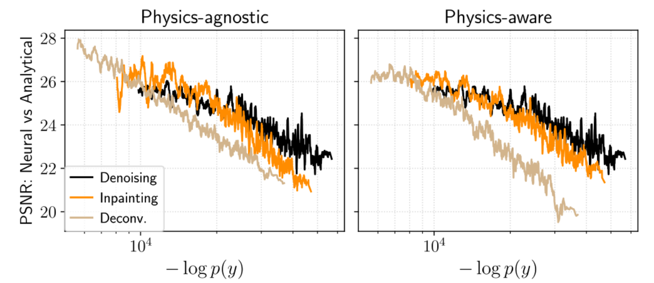


Match analytical/CNN w.r.t. sample density

We capture (no official taxonomy):

- The interpolation regime
- The local generalization regime

The limit of the theory



Match analytical/CNN w.r.t. sample density

We capture (no official taxonomy):

- The interpolation regime
- The local generalization regime

We do not capture the full generalization regime.

Towards a recovery theory

MMSE and sparsity

Remember that $\hat{x}_{MMSE}(y) = \sum_{d=1}^{D} x_d w_d$ with $\mathcal{D} = \{x_1, \dots x_D\}$

$$w_d \propto \exp\left(-\frac{r_d^2}{2\sigma^2}\right) \quad \text{and} \quad r_d = \|Ax_d - y\|$$

MMSE and sparsity

Remember that $\hat{x}_{MMSE}(y) = \sum_{d=1}^{D} x_d w_d$ with $\mathcal{D} = \{x_1, \ldots x_D\}$

$$w_d \propto \exp\left(-\frac{r_d^2}{2\sigma^2}\right) \quad \text{and} \quad r_d = \|Ax_d - y\|$$

The general intuition

Assume $r_1 \leq r_2 \leq \ldots \leq r_D$. Weights w_d are very sparse!

$$w_1 \geq w_2 \geq \ldots \geq w_D \overset{\text{divide by } w_1}{\iff} 1 \geq \exp\left(-\frac{r_2^2 - r_1^2}{2\sigma^2}\right) \geq \exp\left(-\frac{r_3^2 - r_1^2}{2\sigma^2}\right) \geq \exp\left(-\frac{r_4^2 - r_1^2}{2\sigma^2}\right) \geq$$

If $r_d^2 - r_1^2 \gtrsim \sigma^2$, the contribution of x_d is negligible

Typically $\|x_d - x_{d'}\| \gg 1$ is since we are in high dimension

Then we expect $\|A(x_d - x_{d'})\| \gg 1$ as well

MMSE and sparsity

Remember that $\hat{x}_{MMSE}(y) = \sum_{d=1}^{D} x_d w_d$ with $\mathcal{D} = \{x_1, \dots x_D\}$

$$w_d \propto \exp\left(-\frac{r_d^2}{2\sigma^2}\right) \quad \text{and} \quad r_d = \|Ax_d - y\|$$

The general intuition

Assume $r_1 \leq r_2 \leq \dots \leq r_D$. Weights w_d are very sparse!

$$w_1 \geq w_2 \geq \dots \geq w_D \overset{\text{divide by } w_1}{\Longleftrightarrow} 1 \geq \exp\left(-\frac{r_2^2 - r_1^2}{2\sigma^2}\right) \geq \exp\left(-\frac{r_3^2 - r_1^2}{2\sigma^2}\right) \geq \exp\left(-\cdots\right) \geq$$

If $r_d^2 - r_1^2 \gtrsim \sigma^2$, the contribution of x_d is negligible

Typically $\|x_d - x_{d'}\| \gg 1$ is since we are in high dimension

Then we expect $\|A(x_d - x_{d'})\| \gg 1$ as well

We can make this idea precise...

Let $\bar{x} \in \mathbb{R}^N$ and set $\alpha(\bar{x}) \stackrel{\text{def.}}{=} \inf_{x \in \mathcal{D}} \frac{\|A(x-\bar{x})\|}{\|x-\bar{x}\|}$.

$\alpha(\bar{x}) \approx$ restricted injectivity constant (can be nonzero even for $\mathcal{D}$ manifold)

Theorem (to appear)

Assume $D \geq 3$, $\|e\| \leq \varepsilon$ and set $r_* \stackrel{\text{def.}}{=} \min_{x \in \mathcal{D}} \|A(x-\bar{x})\|$

$$\text{Then} \quad \|\hat{x}_{MMSE}(y) - \bar{x}\| \leq \frac{1}{\alpha(\bar{x})} \left(\underbrace{r_*}_{best} + \underbrace{2\varepsilon}_{variance} + \underbrace{\sigma\sqrt{2\ln(D)}}_{bias} \right).$$

Let $\bar{x} \in \mathbb{R}^N$ and set $\alpha(\bar{x}) \stackrel{\text{def.}}{=} \inf_{x \in \mathcal{D}} \frac{\|A(x-\bar{x})\|}{\|x-\bar{x}\|}$.

$\alpha(\bar{x}) \approx$ restricted injectivity constant (can be nonzero even for $\mathcal{D}$ manifold)

Theorem (to appear)

Assume $D \geq 3$, $\|e\| \leq \varepsilon$ and set $r_* \stackrel{\text{def.}}{=} \min_{x \in \mathcal{D}} \|A(x-\bar{x})\|$

$$\text{Then} \quad \|\hat{x}_{MMSE}(y) - \bar{x}\| \leq \frac{1}{\alpha(\bar{x})} \left(\underbrace{r_*}_{best} + \underbrace{2\varepsilon}_{variance} + \underbrace{\sigma\sqrt{2\ln(D)}}_{bias} \right).$$

Interpretation

- r_*: distance database-signal in measurement space
- 2ε: *variance*, effect of noise
- $\sigma\sqrt{2\ln(D)}$: *ridiculous bias*, effect of points beyond r_1

$$D = 10^6 \rightsquigarrow 5\sigma \qquad \text{and} \qquad D = 10^{24} \rightsquigarrow 10\sigma$$

Let $\bar{x} \in \mathbb{R}^N$ and set $\alpha(\bar{x}) \overset{\text{def.}}{=} \inf_{x \in \mathcal{D}} \frac{\|A(x - \bar{x})\|}{\|x - \bar{x}\|}$.

$\alpha(\bar{x}) \approx$ restricted injectivity constant (can be nonzero even for $\mathcal{D}$ manifold)

Theorem (to appear)

Assume $D \geq 3$, $\|e\| \leq \varepsilon$ and set $r_* \overset{\text{def.}}{=} \min_{x \in \mathcal{D}} \|A(x - \bar{x})\|$

$$\text{Then} \quad \|\hat{x}_{MMSE}(y) - \bar{x}\| \leq \frac{1}{\alpha(\bar{x})} \left(\underbrace{r_*}_{best} + \underbrace{2\varepsilon}_{variance} + \underbrace{\sigma\sqrt{2\ln(D)}}_{bias} \right).$$

Interpretation

- r_*: distance database-signal in measurement space
- 2ε: *variance*, effect of noise
- $\sigma\sqrt{2\ln(D)}$: *ridiculous bias*, effect of points beyond r_1

$$D = 10^6 \rightsquigarrow 5\sigma \qquad \text{and} \qquad D = 10^{24} \rightsquigarrow 10\sigma$$

Currently extending to local CNNs

Blind inverse problems & DeepBlur

The framework

Assume

$$y = \bar{A}\bar{x} + e$$

both $\bar{A}$ and $\bar{x}$ unknown

Extension to blind inverse problems

The framework

Assume

$$y = \bar{A}\bar{x} + e$$

both $\bar{A}$ and $\bar{x}$ unknown

Training

Given $\mathcal{A} = \{A_1, \ldots, A_P\}$ and $\mathcal{D} = \{x_1, \ldots, x_D\}$, construct

$$\boldsymbol{y} = \mathrm{A}\boldsymbol{x} + \boldsymbol{e} \quad \text{with} \quad \mathrm{A} \sim p_{\mathcal{A}}, \boldsymbol{x} \sim p_{\mathcal{D}}$$

Then for an MLP:

$$\hat{A}_{MMSE}(y) \propto \sum_{A \in \mathcal{A}, x \in \mathcal{D}} A \exp\left(-\frac{\|Ax - y\|^2}{2\sigma^2}\right)$$

Now extending the recovery theory to this setting

Example with blind deblurring

Diffraction blurs: Fresnel approximation

The 3D PSF is given by the pupil function p:

$$k(x, y, z) = \left| \int p(w_1, w_2) \cdot \exp(2i\pi(w_1 x + w_2 y)) \cdot \exp(2i\pi z \cdot d(w_1, w_2)) dw_1 dw_2 \right|^2$$

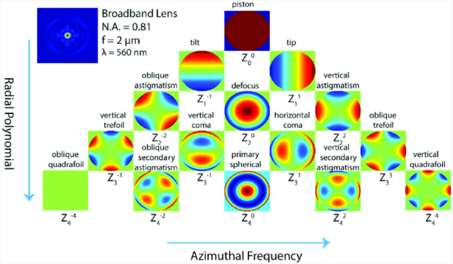
Example with blind deblurring

Diffraction blurs: Fresnel approximation

The 3D PSF is given by the pupil function p:

$$k(x, y, z) = \left| \int p(w_1, w_2) \cdot \exp(2i\pi(w_1 x + w_2 y)) \cdot \exp(2i\pi z \cdot d(w_1, w_2)) dw_1 dw_2 \right|^2$$

Approximate the pupil with Zernike polynomials: $p \simeq \exp\left(2i \sum_{n=1}^{N} \gamma_p Z_p\right)$.

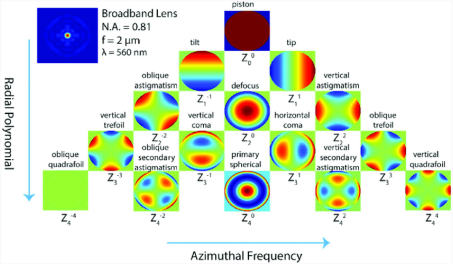


Diffraction blurs: Fresnel approximation

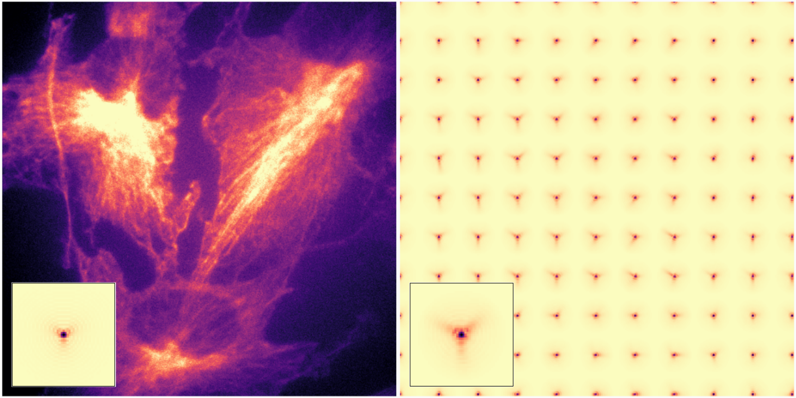
The 3D PSF is given by the pupil function p:

$$k(x, y, z) = \left| \int p(w_1, w_2) \cdot \exp(2i\pi(w_1 x + w_2 y)) \cdot \exp(2i\pi z \cdot d(w_1, w_2)) dw_1 dw_2 \right|^2$$

Approximate the pupil with Zernike polynomials: $p \simeq \exp\left(2i \sum_{n=1}^{N} \gamma_p Z_p\right)$.



Example with blind deblurring: DeepBlur



Comparing an induced PSF with an SLM to the DeepBlur result

Take Home

- Analytical formula for constrained MMSE
- Good match with CNNs in the "local generalization" regime
- Theory helps understanding many non intuitive facts
  - Large datasets are better (more spread p_y)
  - Physics aware not always better
  - Augmentation $\neq$ equivariance
- Towards a reliable recovery theory?
- AI can be reliable, not a black-box with intuition, reasoning, tests

(a) M.H. Nguyen

(b) E. Pauwels

(c) Q.B. Do