Seminario Optimización y aprendizaje automático

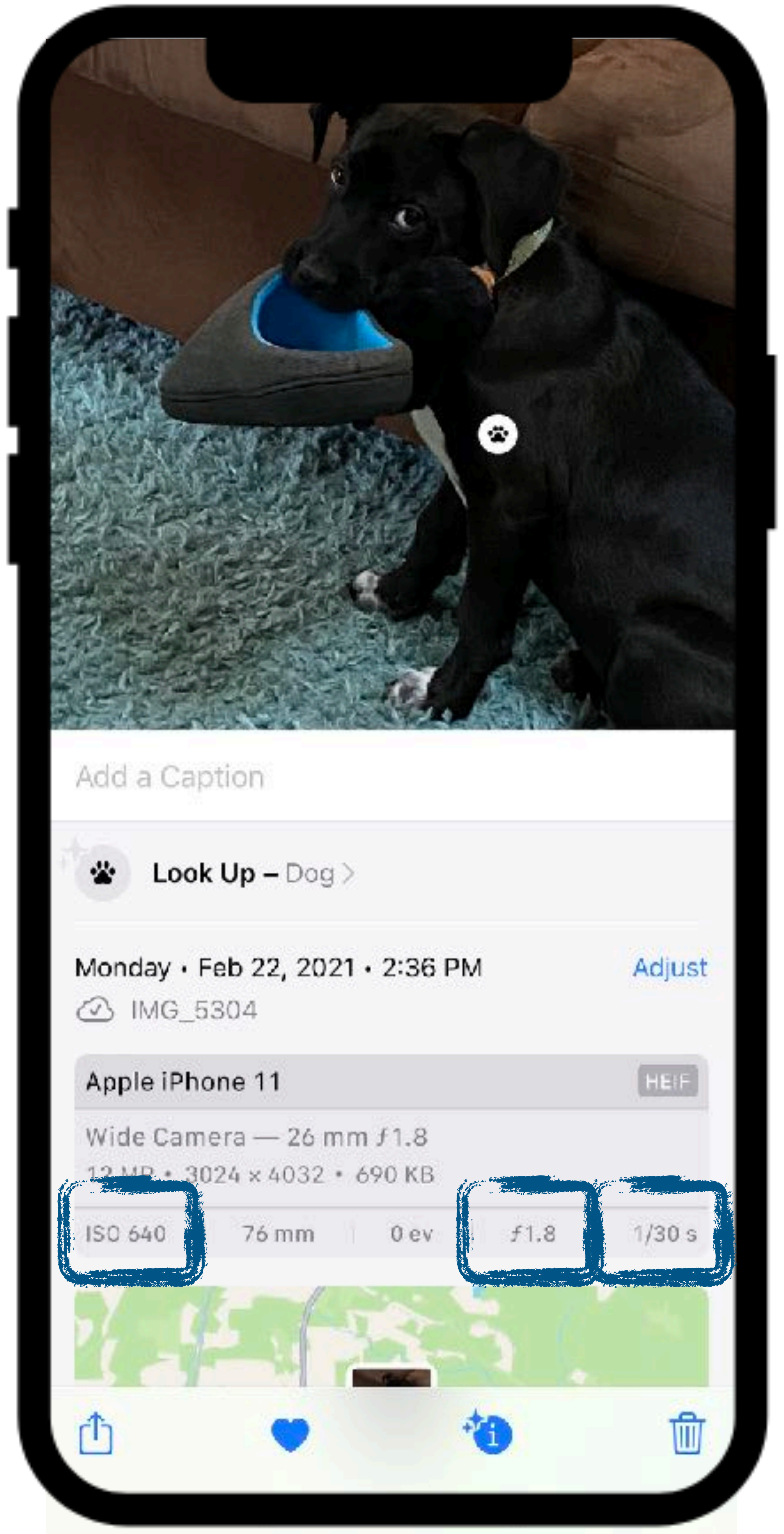
Brief history of denoising and their "nobel" conception as implicit manifold learners

J. Matias Di martino

October 2024

Noise, why I care?

Digital Imag



Add a Caption

Look Up – Dog >

Monday • Feb 22, 2021 • 2:36 PM Adjust

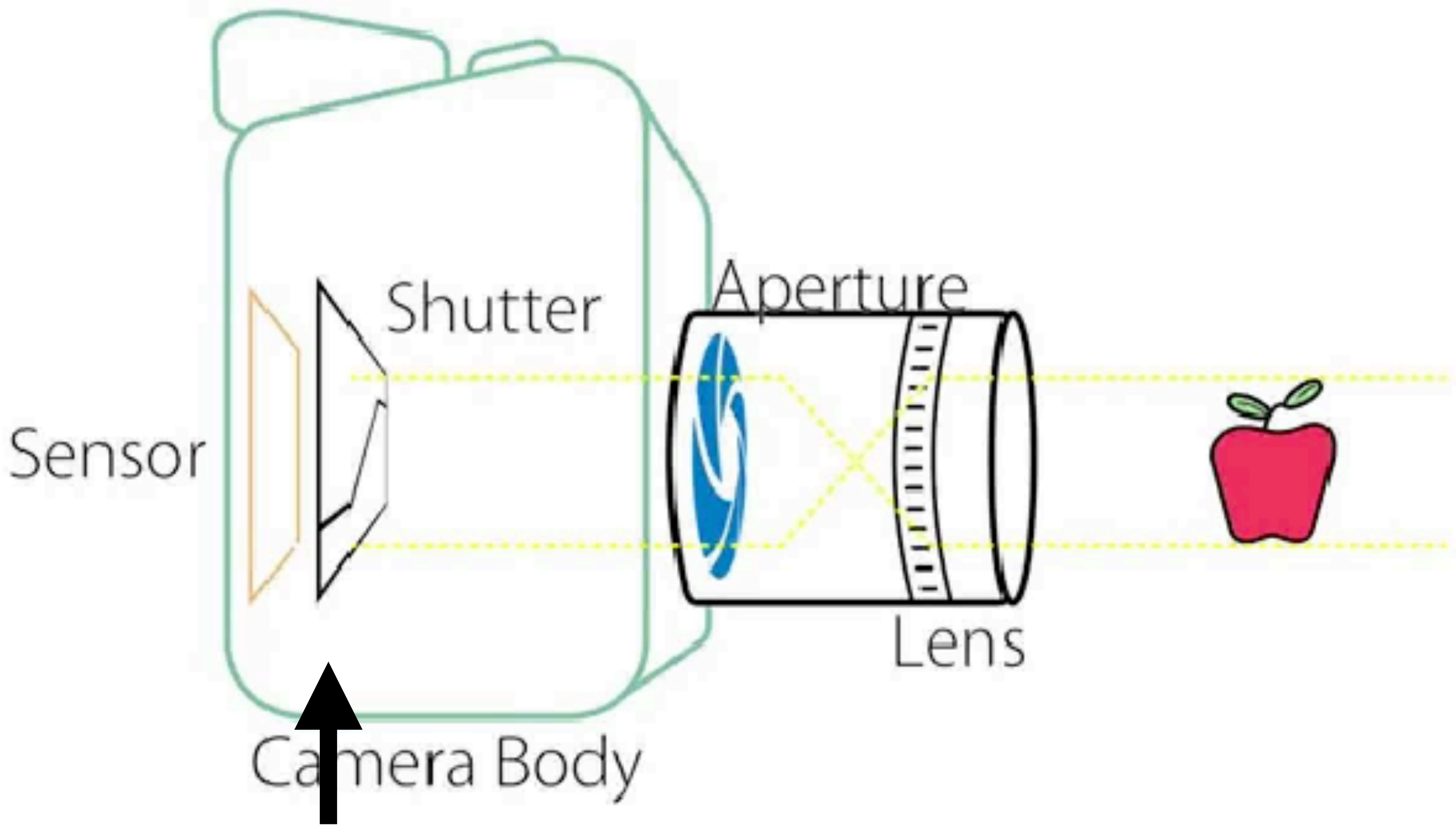
IMG_5304

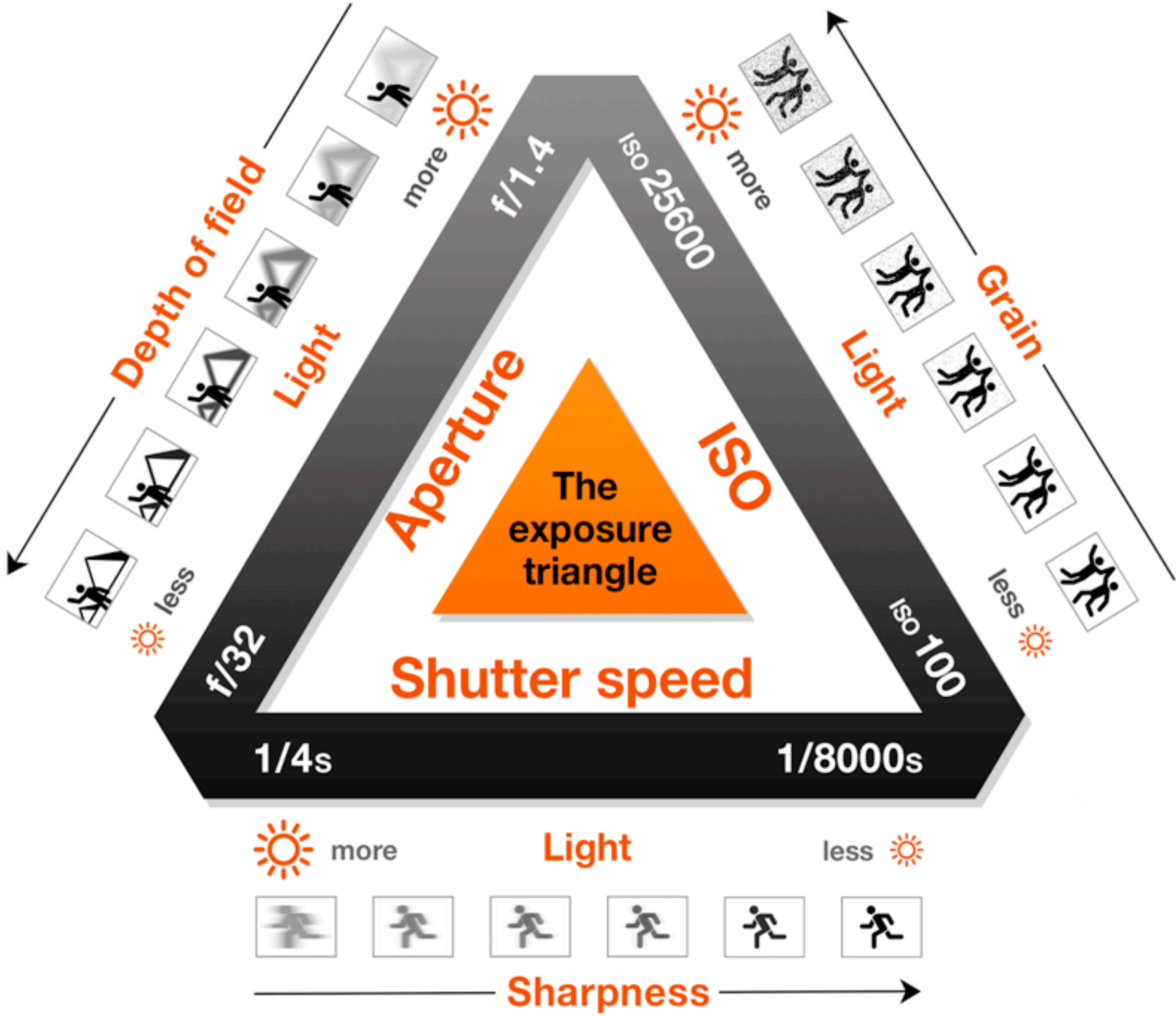
Apple iPhone 11 HEIF

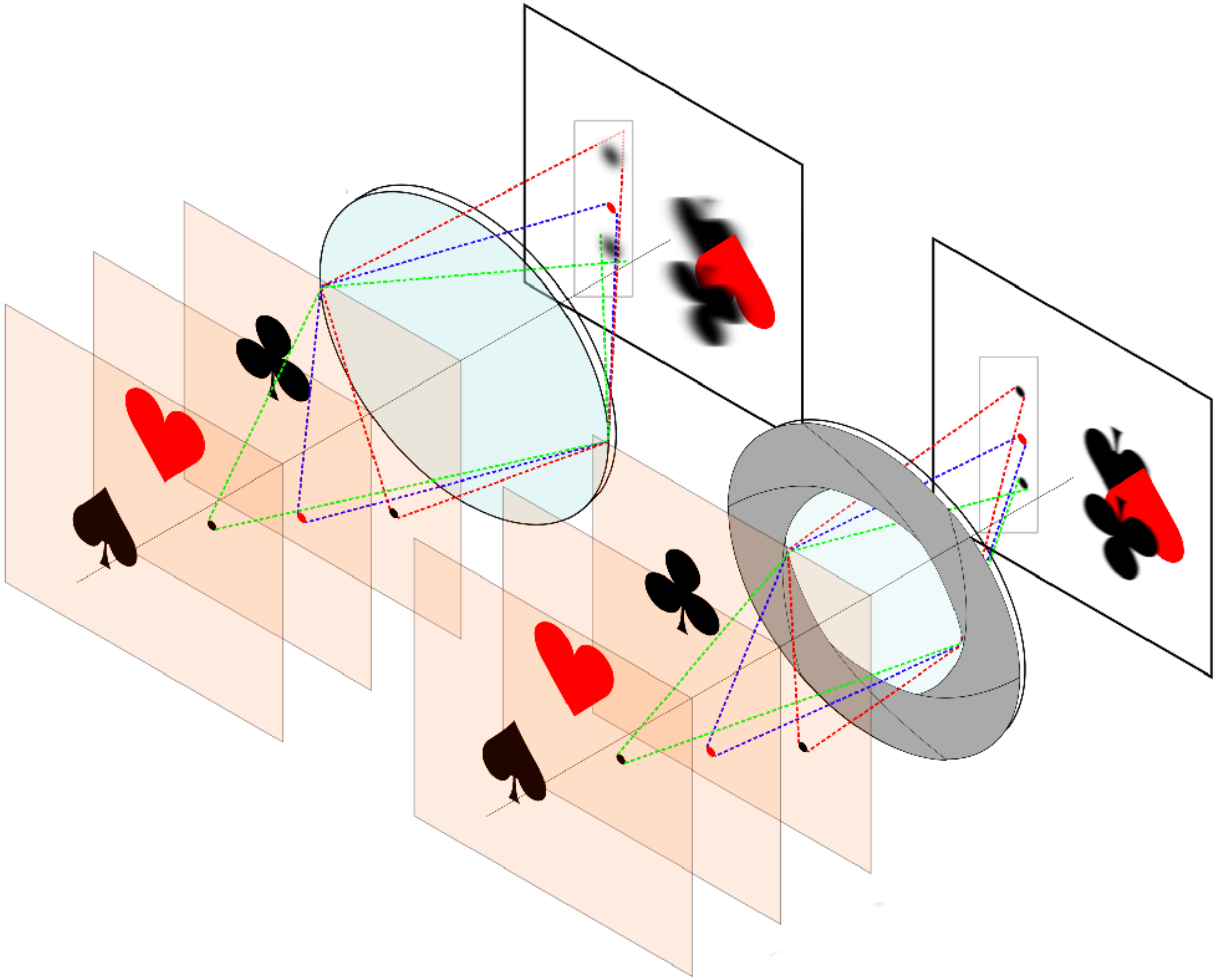
Wide Camera — 26 mm ƒ1.8

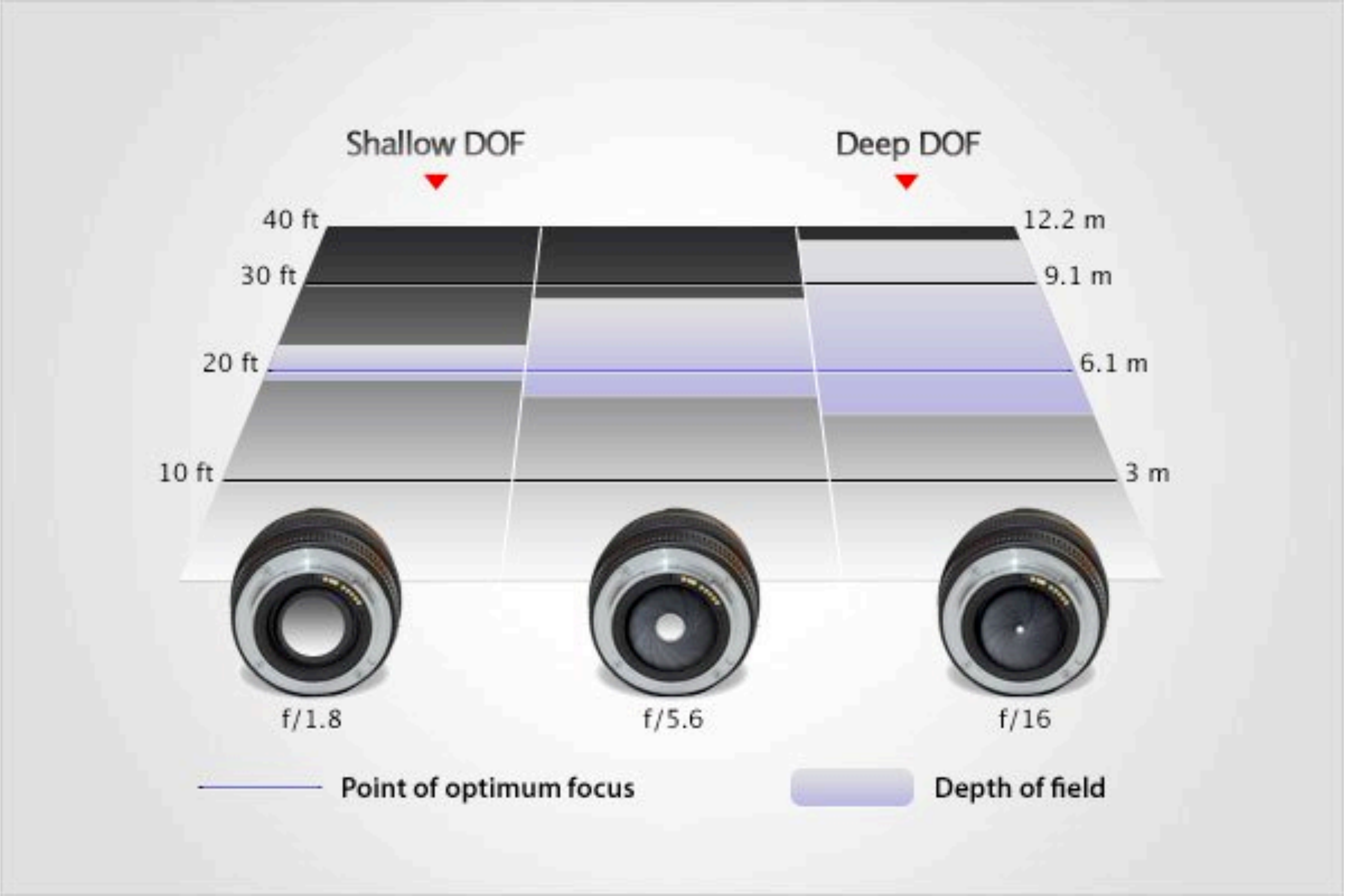
12 MP • 3024 x 4032 • 690 KB

ISO 640 | 76 mm | 0 ev | ƒ1.8 | 1/30 s





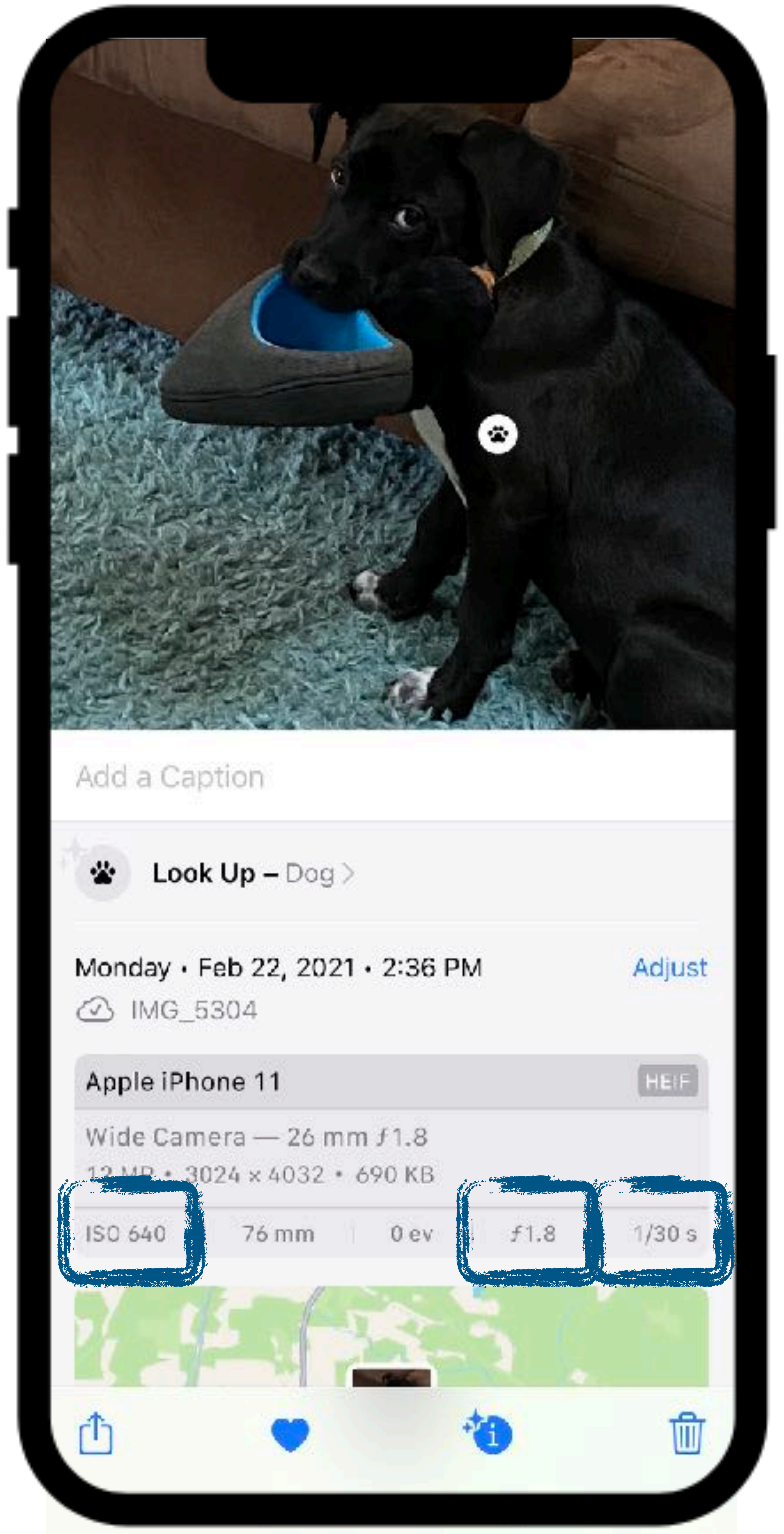




Shallow DOF
Deep DOF
40 ft
30 ft
20 ft
10 ft
12.2 m
9.1 m
6.1 m
3 m
f/1.8
f/5.6
f/16
Point of optimum focus
Depth of field

f/1.4
f/2
f/2.8
f/4
f/5.6
f/8
f/11
f/16

Digital Imag



Add a Caption

Look Up – Dog >

Monday • Feb 22, 2021 • 2:36 PM Adjust

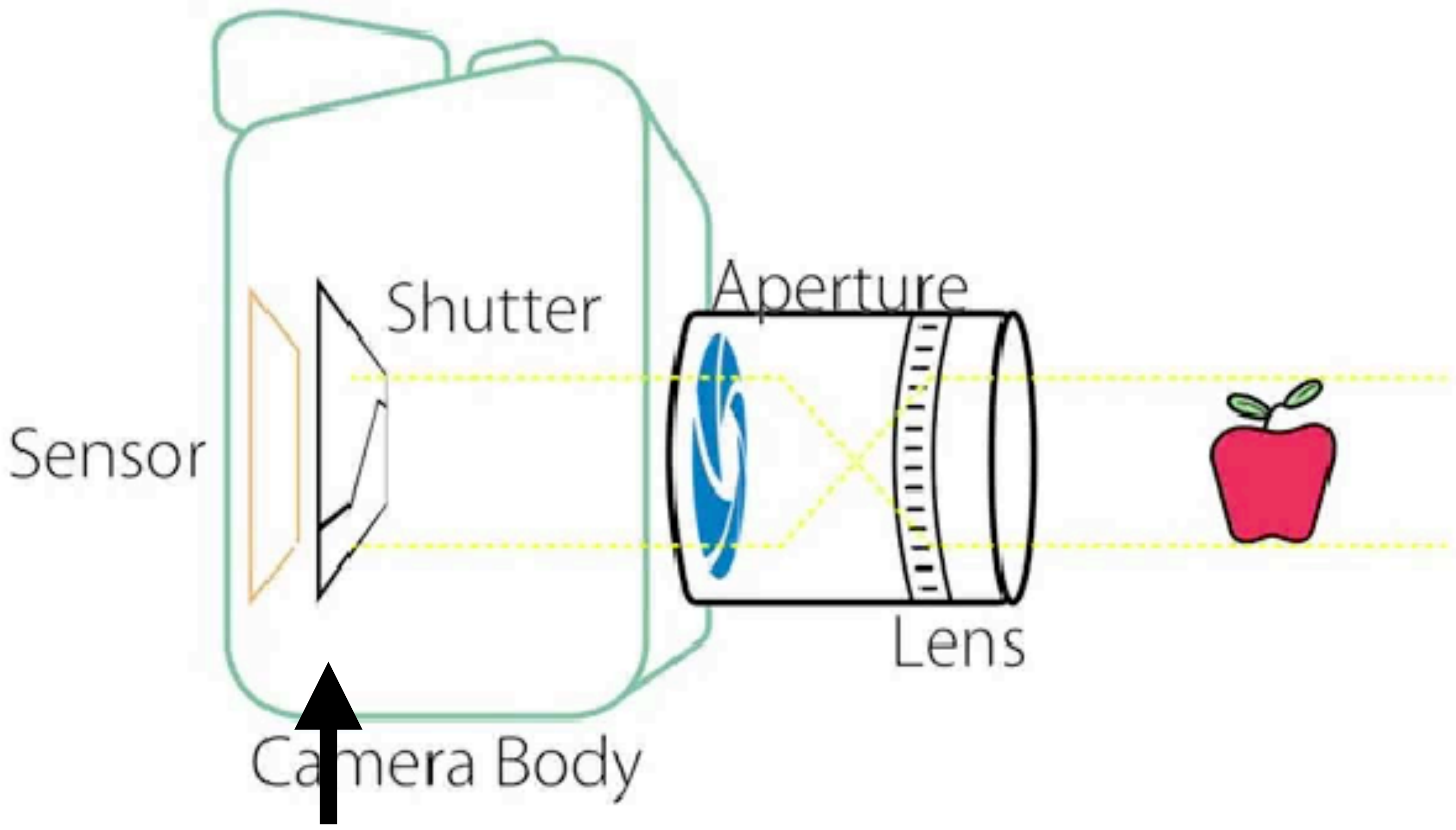
IMG_5304

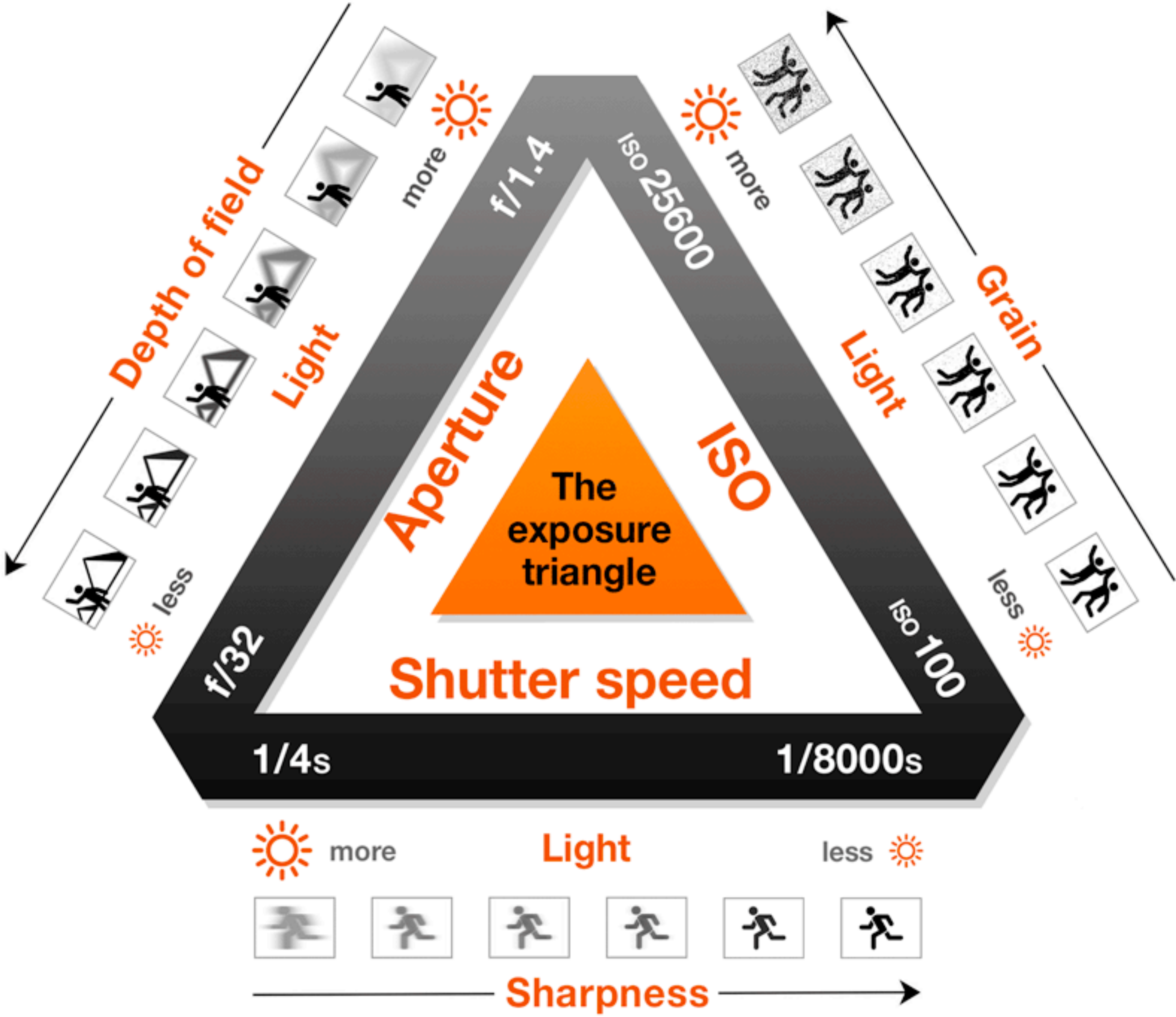
Apple iPhone 11 HEIF

Wide Camera — 26 mm ƒ1.8

12 MP • 3024 × 4032 • 690 KB

ISO 640 | 76 mm | 0 ev | ƒ1.8 | 1/30 s





ISO 250 18-140@18.0 mm f/5 1/3s

ISO 5000
18-140@85 mm
f/5.3
1/20s

ISO 5000 18-140@85 mm f/5.3 1/20s

Cool, how can I remove noise?

$$f(x) = \|x - y\|_2^2 + R(x)$$

$$f(x) = \boxed{\|x - y\|_2^2} + \boxed{R(x)}$$

Fitting Data

Regularization

$$\hat{x} = \text{argmin}_x \|x - y\|_2^2 + R(x)$$

$$f(x) = \|x - y\|_2^2 + \cancel{R(x)}$$

Fitting Data

Regularization

y

$\hat{x} = \mathrm{argmin}\{f(x)\}$

$$f(x) = \cancel{\|x - y\|_2^2} + R(x)$$

Fitting Data

Regularization

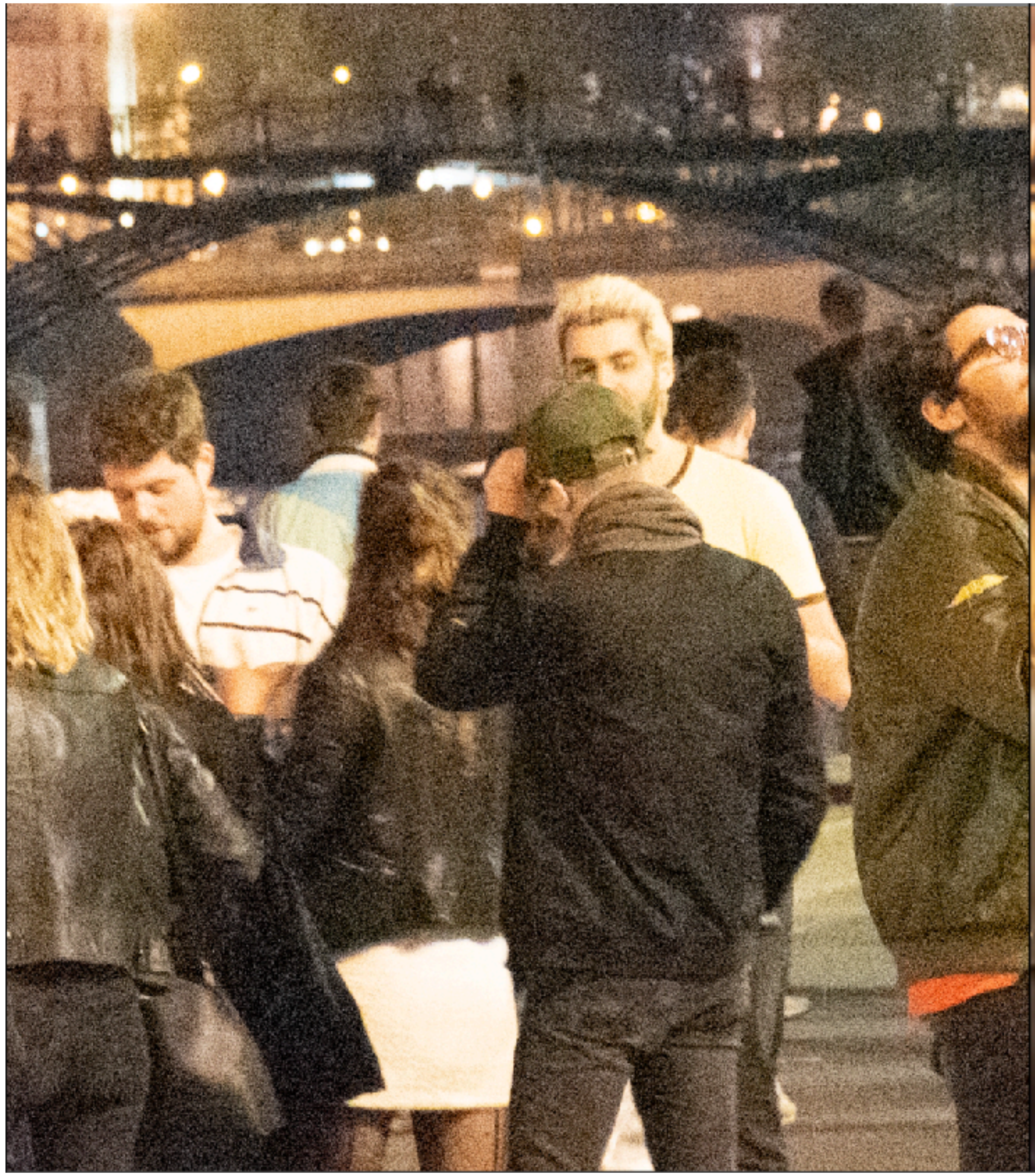
y

$\hat{x} = \mathrm{argmin}\{f(x)\}$

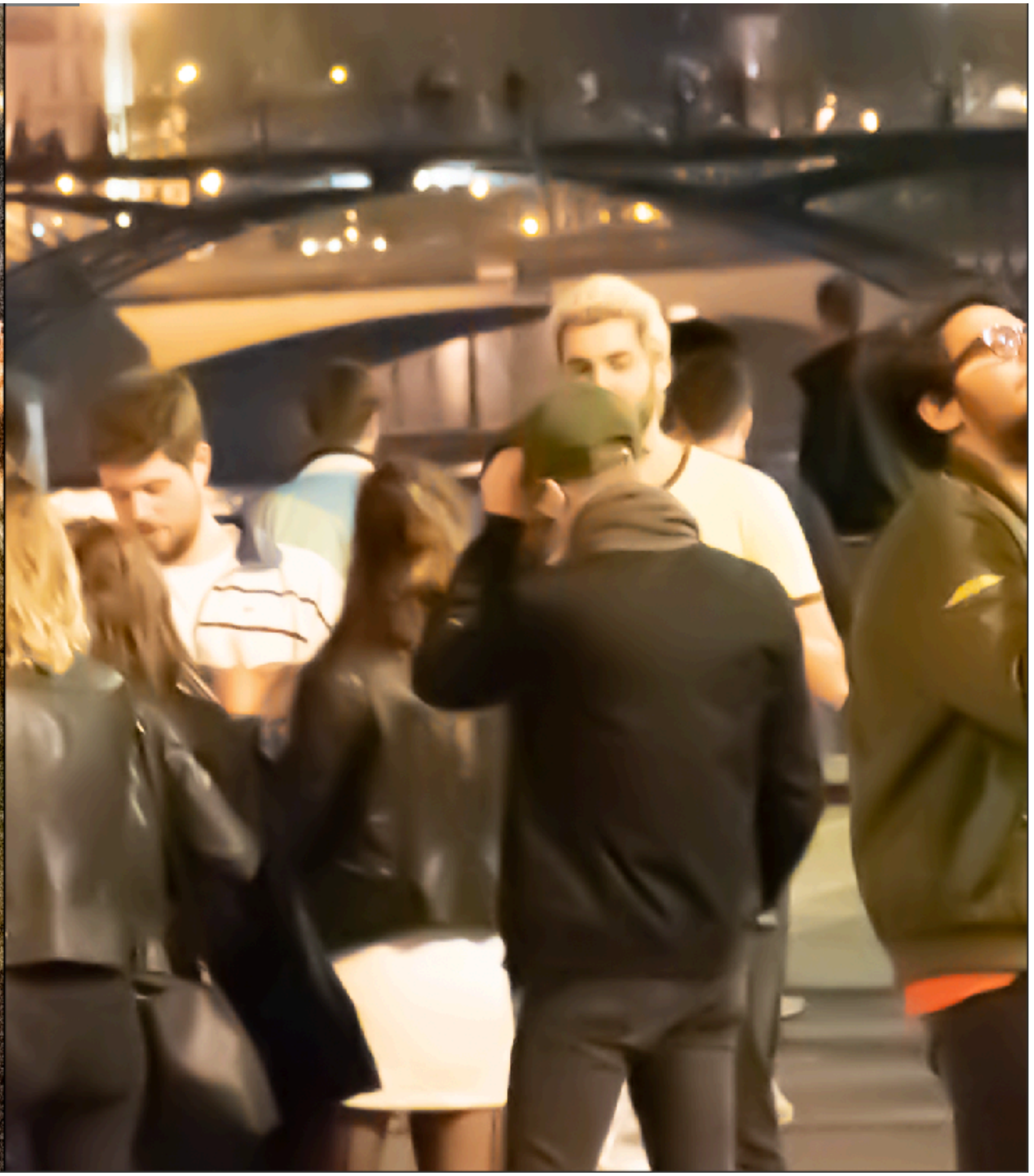
$$f(x) = \|x - y\|_2^2 + R(x)$$

Fitting Data Regularization

y



$\hat{x} = \operatorname{argmin}\{f(x)\}$



$$f(x) = \underbrace{\boxed{\|x - y\|_2^2}}_{\text{Fitting Data}} + \underbrace{\boxed{R(x)}}_{\text{Regularization}}$$

$$f(x) = \|x - y\|_2^2 + R(x)$$

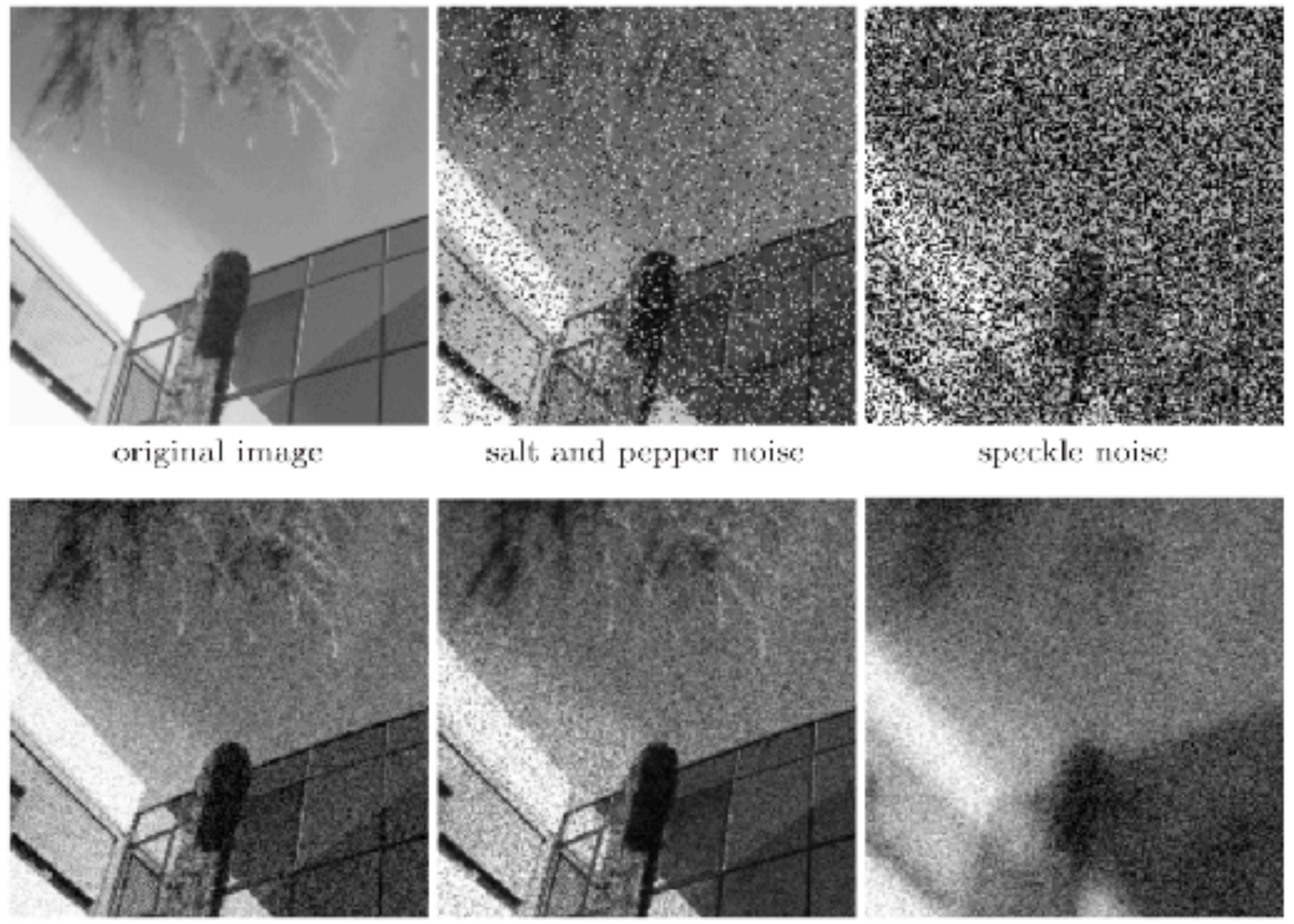
Fitting Data

Regularization

$$x^{n+1} = x^n + \delta\left((x^n - y) + \frac{\partial}{\partial x} G(x) \right)$$

$$\frac{\partial x}{\partial t} = - \partial f \quad (\text{EL equations})$$

for implementation details see: https://nbviewer.org/github/gpeyre/numerical-tours/blob/master/matlab/denoisingsimp_4_denoiseregul.ipynb



original image

salt and pepper noise

speckle noise

additive Gaussian noise

multiplicative Gaussian noise

blur and additive Gaussian noise

3.2.2 Regularization of the Problem

A classical way to overcome ill-posed minimization problems is to add a regularization term to the energy. This idea was introduced in 1977 by Tikhonov and Arsenin [317]. The authors proposed to consider the following minimization problem:

$$F(u) = \int_{\Omega} |u_0 - Ru|^2 \; dx + \lambda \int_{\Omega} |\nabla u|^2 \; dx. \tag{3.4}$$

Aubert, G. & Kornprobst, P. Mathematical Problems in Image Processing Partial Differential Equations and the Calculus of Variations. (2006). doi:10.2307/3615195.

$$\lim_{\varepsilon \to 0} \frac{E[u+\varepsilon v]-E[u]}{\varepsilon} = \lim_{\varepsilon \to 0} \frac{1}{\varepsilon} \int \underbrace{(\nabla u + \varepsilon \nabla v)^T (\nabla u + \varepsilon \nabla v)} - \underbrace{\nabla u^T \nabla u}$$

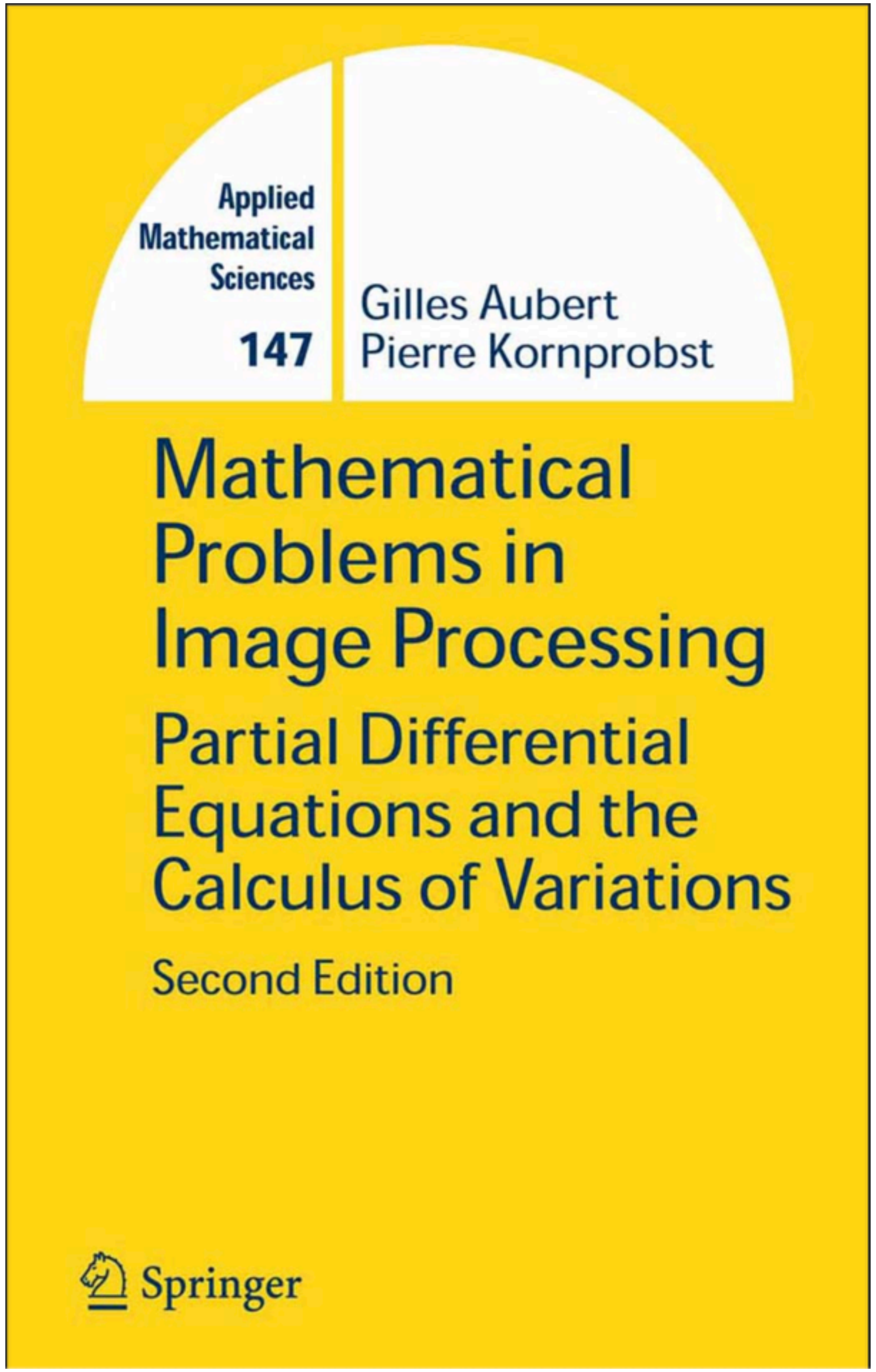
$$\lim_{\varepsilon \to 0} \frac{1}{\varepsilon} \int \overbrace{\cancel{\nabla u^T \nabla u} + \underbrace{\nabla u^T \varepsilon \nabla v} + \underbrace{\varepsilon \nabla v^T \nabla u} + \varepsilon^2 \nabla v^T \nabla v} - \overbrace{\cancel{\nabla u^T \nabla u}}$$

$$= \lim_{\varepsilon \to 0} \frac{1}{\varepsilon} \int \left(2 \varepsilon \nabla v^T \nabla u + \varepsilon^2 |\nabla v|^2 \right) d\Omega$$

$$= \lim_{\varepsilon \to 0} \cancel{\frac{1}{\varepsilon}} \int \left(2 \cancel{\varepsilon} \nabla v^T \nabla u + \varepsilon^{\cancel{2}} |\nabla v|^2 \right) d\Omega$$

$$= \int 2 \nabla v^T \nabla u \, d\Omega = -\int 2 v \, \mathrm{div}(\nabla u) \, d\Omega = 0 \ \forall v$$

$$\iff -\mathrm{div}(\nabla u) = 0 = \delta E$$



Applied Mathematical Sciences 147

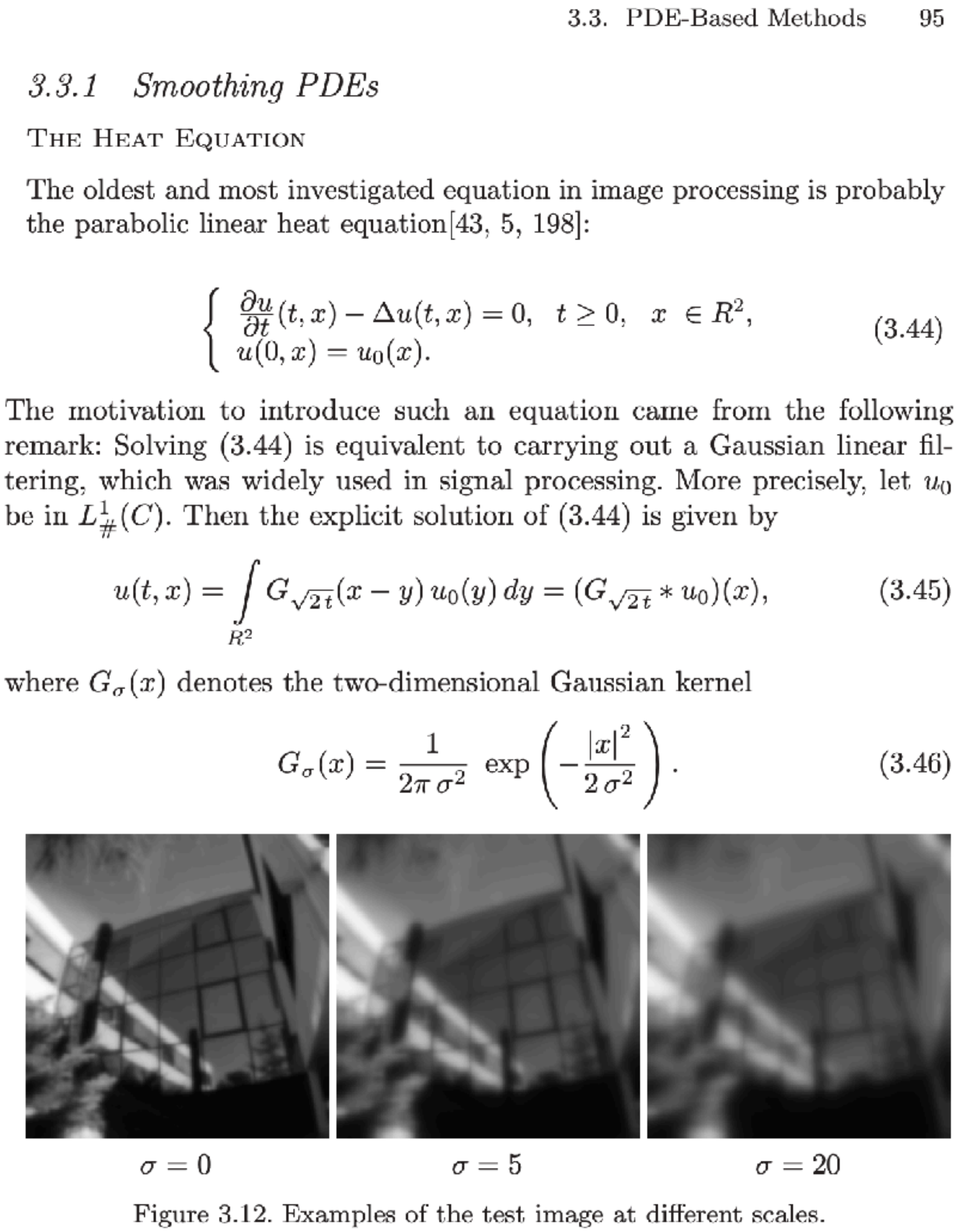
Gilles Aubert
Pierre Kornprobst

Mathematical Problems in Image Processing

Partial Differential Equations and the Calculus of Variations

Second Edition

Springer



3.3. PDE-Based Methods 95

3.3.1 Smoothing PDEs

THE HEAT EQUATION

The oldest and most investigated equation in image processing is probably the parabolic linear heat equation[43, 5, 198]:

$$\begin{cases} \dfrac{\partial u}{\partial t}(t,x) - \Delta u(t,x) = 0, \quad t \geq 0, \quad x \in R^2, \\ u(0,x) = u_0(x). \end{cases} \tag{3.44}$$

The motivation to introduce such an equation came from the following remark: Solving (3.44) is equivalent to carrying out a Gaussian linear filtering, which was widely used in signal processing. More precisely, let u_0 be in $L^1_\#(C)$. Then the explicit solution of (3.44) is given by

$$u(t,x) = \int_{R^2} G_{\sqrt{2t}}(x-y)\, u_0(y)\, dy = (G_{\sqrt{2t}} * u_0)(x), \tag{3.45}$$

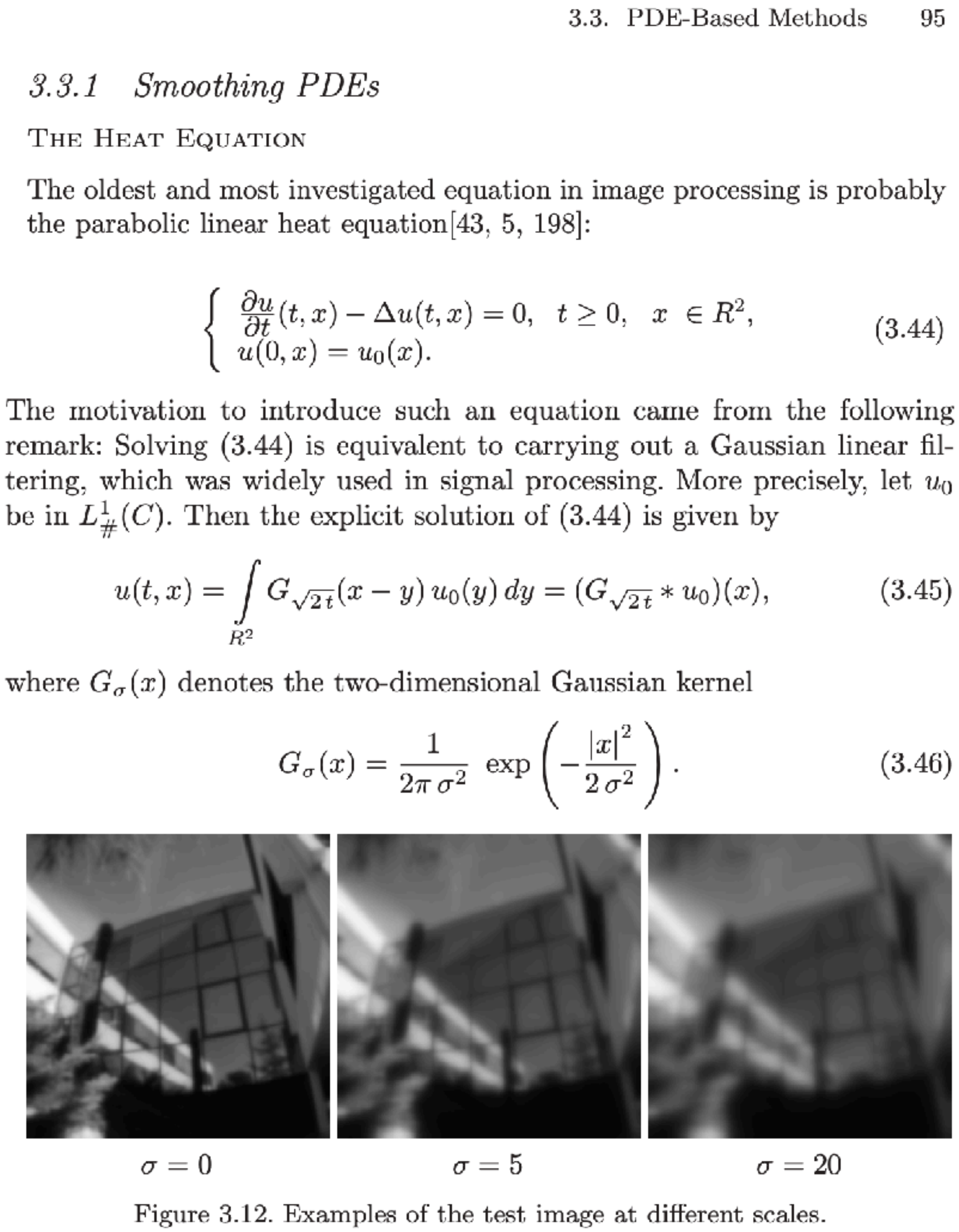
where $G_\sigma(x)$ denotes the two-dimensional Gaussian kernel

$$G_\sigma(x) = \frac{1}{2\pi\,\sigma^2}\ \exp\left(-\frac{|x|^2}{2\,\sigma^2}\right). \tag{3.46}$$

$\sigma = 0$ $\sigma = 5$ $\sigma = 20$

Figure 3.12. Examples of the test image at different scales.

Problem, it blurs uniformly and destroy edges and image contrast



3.3. PDE-Based Methods 95

3.3.1 Smoothing PDEs

The Heat Equation

The oldest and most investigated equation in image processing is probably the parabolic linear heat equation[43, 5, 198]:

$$\begin{cases} \dfrac{\partial u}{\partial t}(t,x) - \Delta u(t,x) = 0, \quad t \geq 0, \quad x \in R^2, \\ u(0,x) = u_0(x). \end{cases} \tag{3.44}$$

The motivation to introduce such an equation came from the following remark: Solving (3.44) is equivalent to carrying out a Gaussian linear filtering, which was widely used in signal processing. More precisely, let u_0 be in $L^1_{\#}(C)$. Then the explicit solution of (3.44) is given by

$$u(t,x) = \int\limits_{R^2} G_{\sqrt{2t}}(x-y)\, u_0(y)\, dy = (G_{\sqrt{2t}} * u_0)(x), \tag{3.45}$$

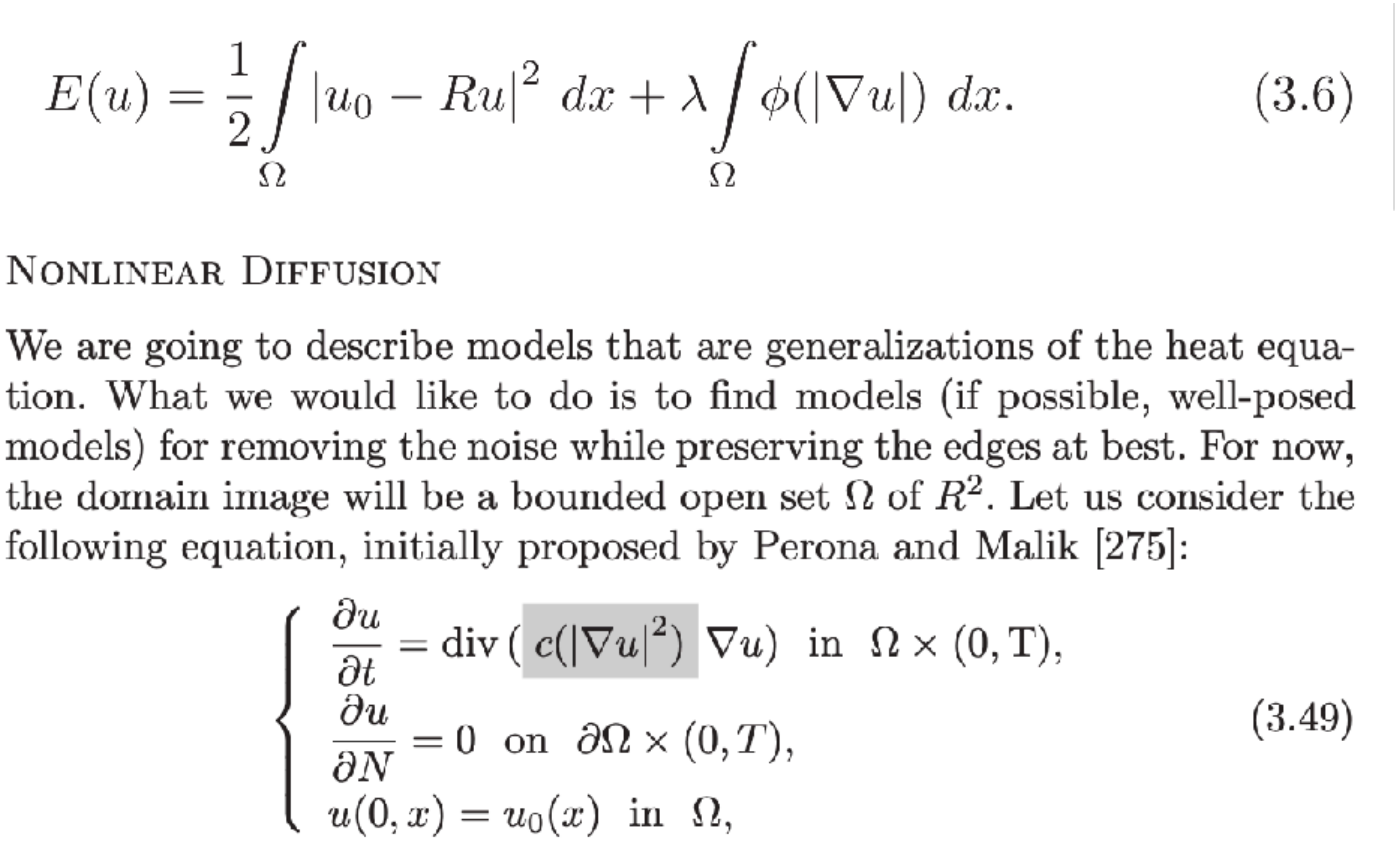
where $G_\sigma(x)$ denotes the two-dimensional Gaussian kernel

$$G_\sigma(x) = \frac{1}{2\pi\,\sigma^2} \exp\left(-\frac{|x|^2}{2\,\sigma^2}\right). \tag{3.46}$$

$\sigma = 0$ $\sigma = 5$ $\sigma = 20$

Figure 3.12. Examples of the test image at different scales.

Solution: Non-linear Diffusion or Non-local Means

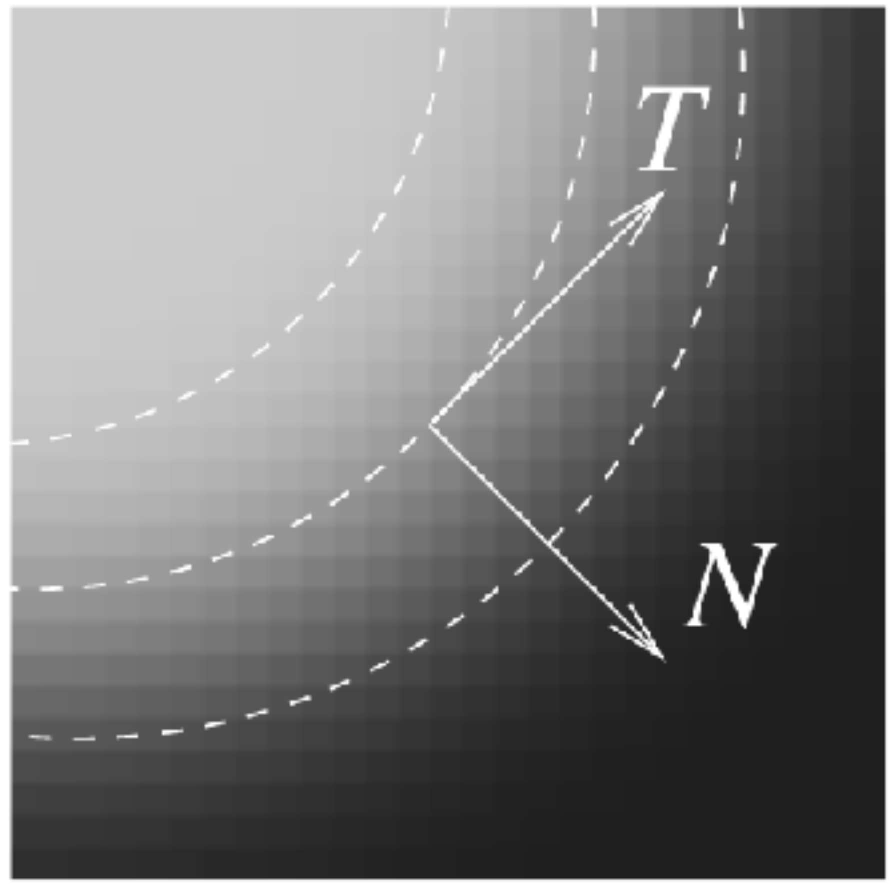


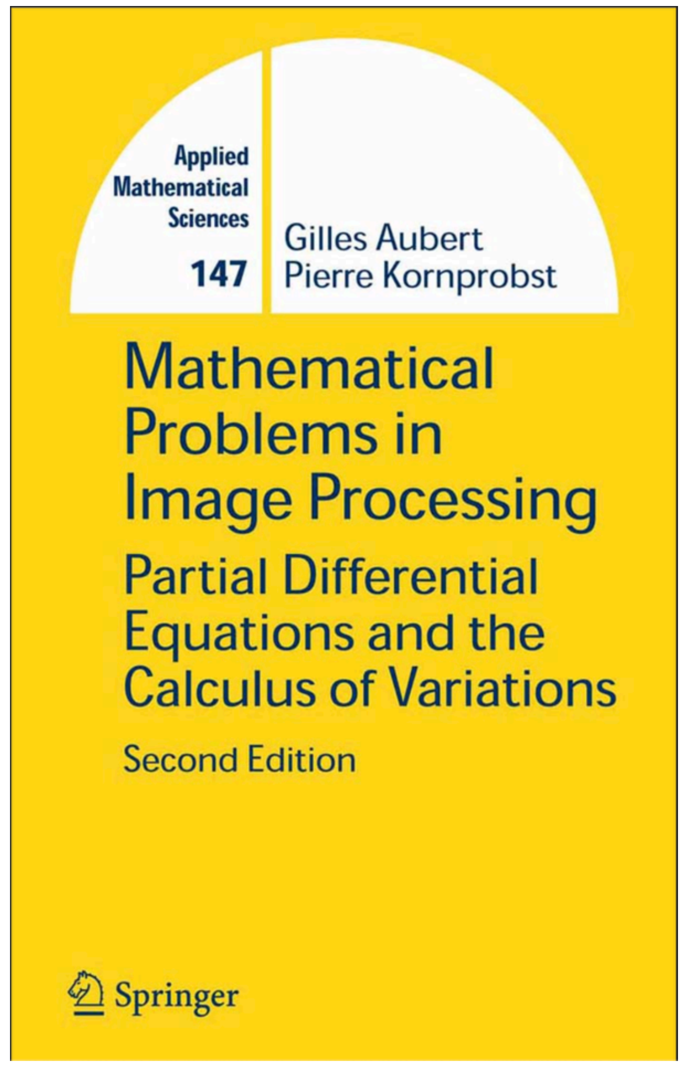
$$E(u) = \frac{1}{2}\int_{\Omega} |u_0 - Ru|^2 \; dx + \lambda \int_{\Omega} \phi(|\nabla u|) \; dx. \tag{3.6}$$

Nonlinear Diffusion

We are going to describe models that are generalizations of the heat equation. What we would like to do is to find models (if possible, well-posed models) for removing the noise while preserving the edges at best. For now, the domain image will be a bounded open set Ω of R^2. Let us consider the following equation, initially proposed by Perona and Malik [275]:

$$\begin{cases} \dfrac{\partial u}{\partial t} = \mathrm{div}\left(c(|\nabla u|^2) \; \nabla u \right) \; \text{ in } \; \Omega \times (0, \mathrm{T}), \\ \dfrac{\partial u}{\partial N} = 0 \; \text{ on } \; \partial\Omega \times (0, T), \\ u(0, x) = u_0(x) \; \text{ in } \; \Omega, \end{cases} \tag{3.49}$$





Applied Mathematical Sciences
147
Gilles Aubert
Pierre Kornprobst

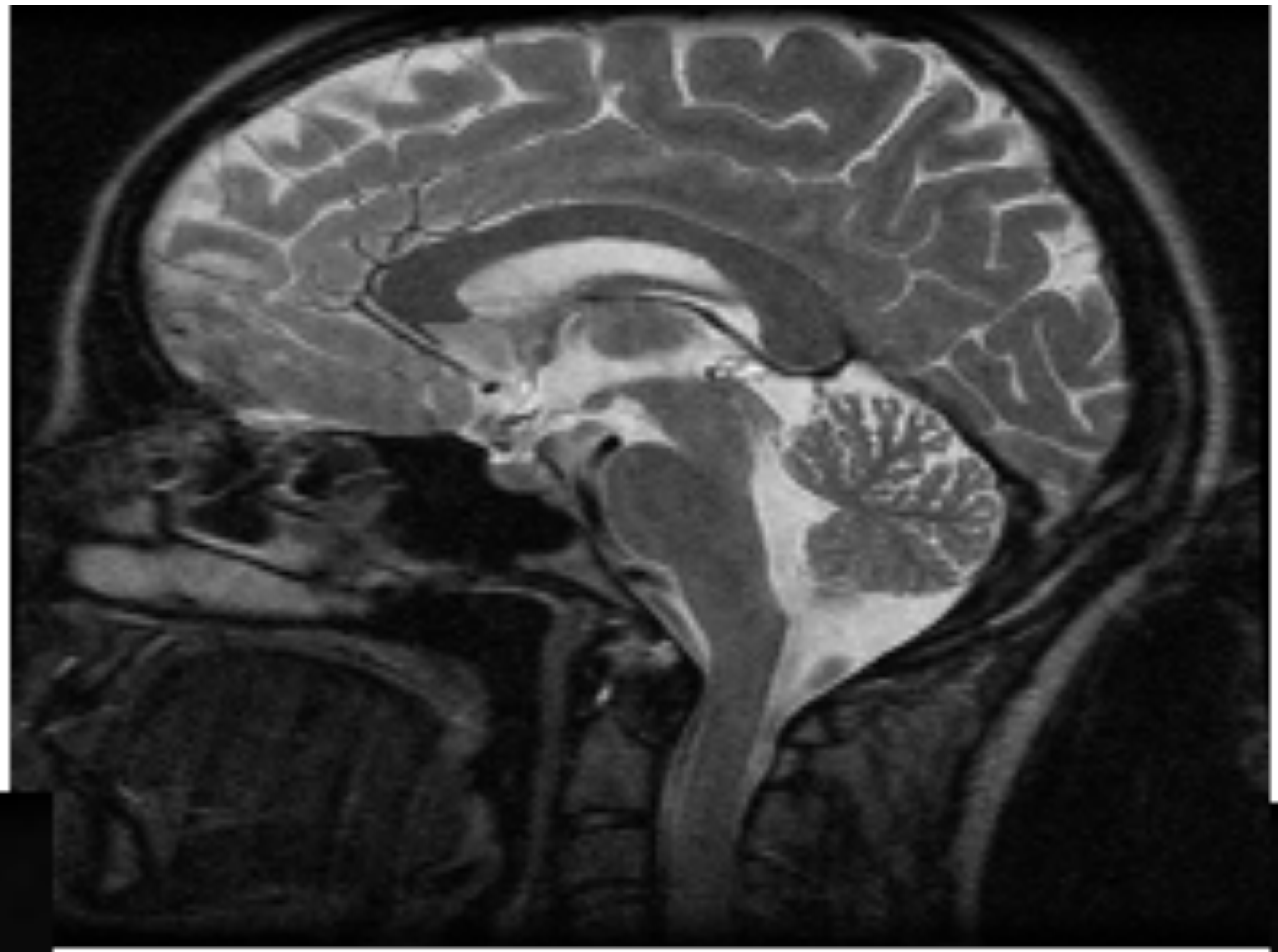
Mathematical Problems in Image Processing

Partial Differential Equations and the Calculus of Variations

Second Edition

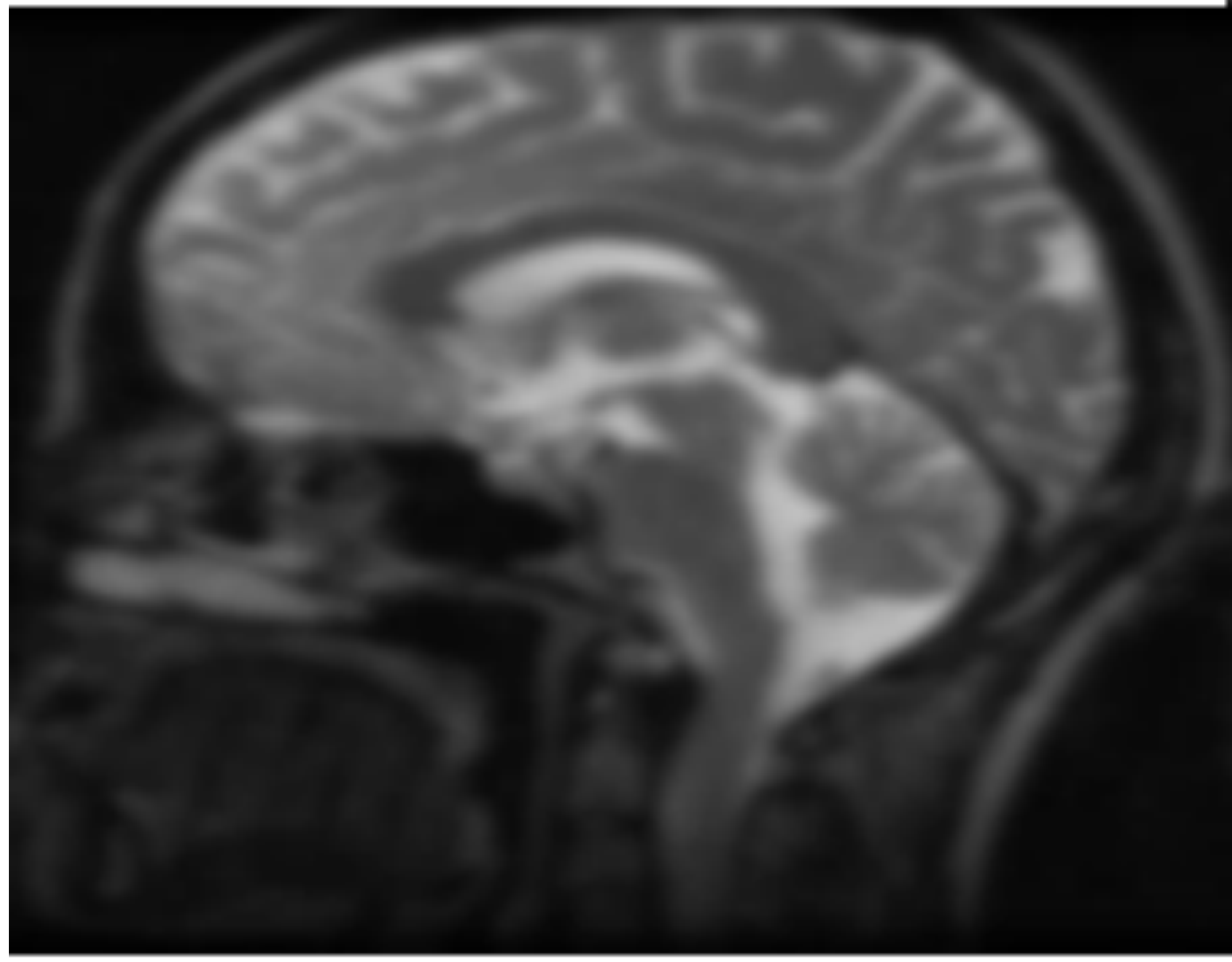
Springer

Original



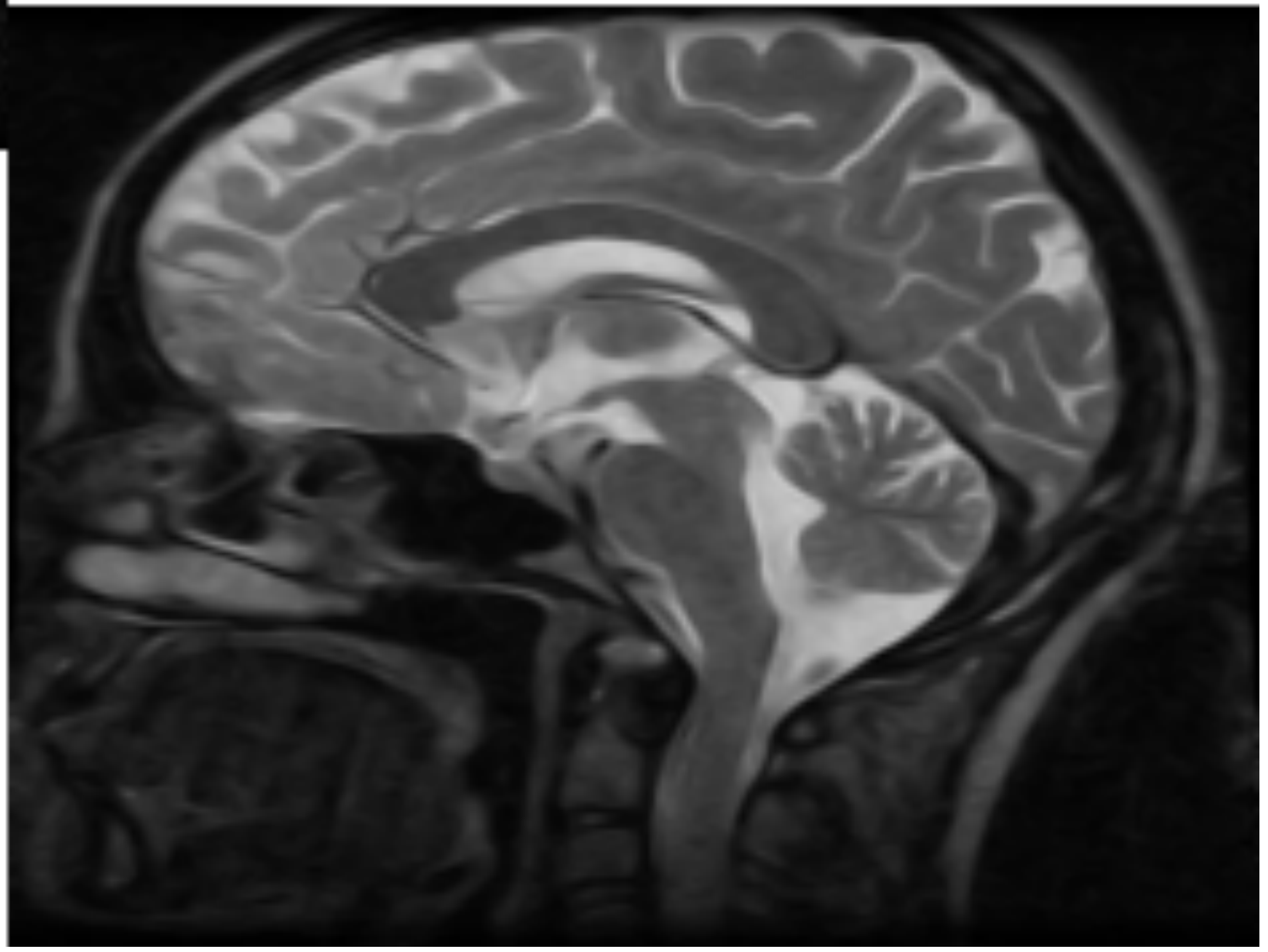
Heat equation

$$E[u] = \int |\nabla u|^2 \to u_t = \Delta u$$



Anisotropic Diffusion

$$\frac{\partial I}{\partial t} = \operatorname{div}\left(c(x,y,t)\nabla I\right)$$



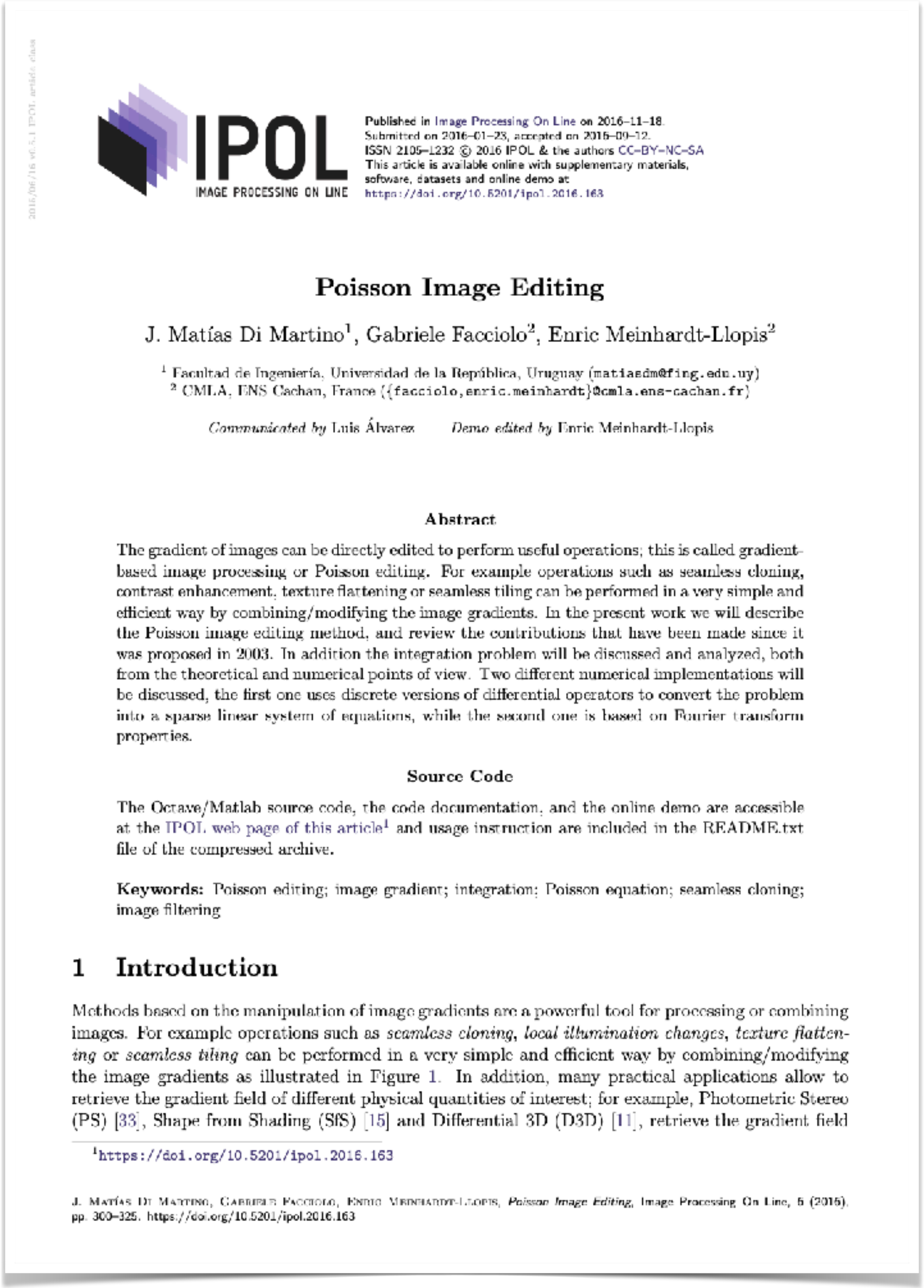


IMAGE PROCESSING ON LINE

Published in Image Processing On Line on 2016–11–18.
Submitted on 2016–01–23, accepted on 2016–09–12.
ISSN 2105–1232 © 2016 IPOL & the authors CC–BY–NC–SA
This article is available online with supplementary materials,
software, datasets and online demo at
https://doi.org/10.5201/ipol.2016.163

Poisson Image Editing

J. Matías Di Martino[1], Gabriele Facciolo[2], Enric Meinhardt-Llopis[2]

[1] Facultad de Ingeniería, Universidad de la República, Uruguay (matiasdm@fing.edu.uy)
[2] CMLA, ENS Cachan, France ({facciolo,enric.meinhardt}@cmla.ens-cachan.fr)

Communicated by Luis Álvarez *Demo edited by* Enric Meinhardt-Llopis

Abstract

The gradient of images can be directly edited to perform useful operations; this is called gradient-based image processing or Poisson editing. For example operations such as seamless cloning, contrast enhancement, texture flattening or seamless tiling can be performed in a very simple and efficient way by combining/modifying the image gradients. In the present work we will describe the Poisson image editing method, and review the contributions that have been made since it was proposed in 2003. In addition the integration problem will be discussed and analyzed, both from the theoretical and numerical points of view. Two different numerical implementations will be discussed, the first one uses discrete versions of differential operators to convert the problem into a sparse linear system of equations, while the second one is based on Fourier transform properties.

Source Code

The Octave/Matlab source code, the code documentation, and the online demo are accessible at the IPOL web page of this article[1] and usage instruction are included in the README.txt file of the compressed archive.

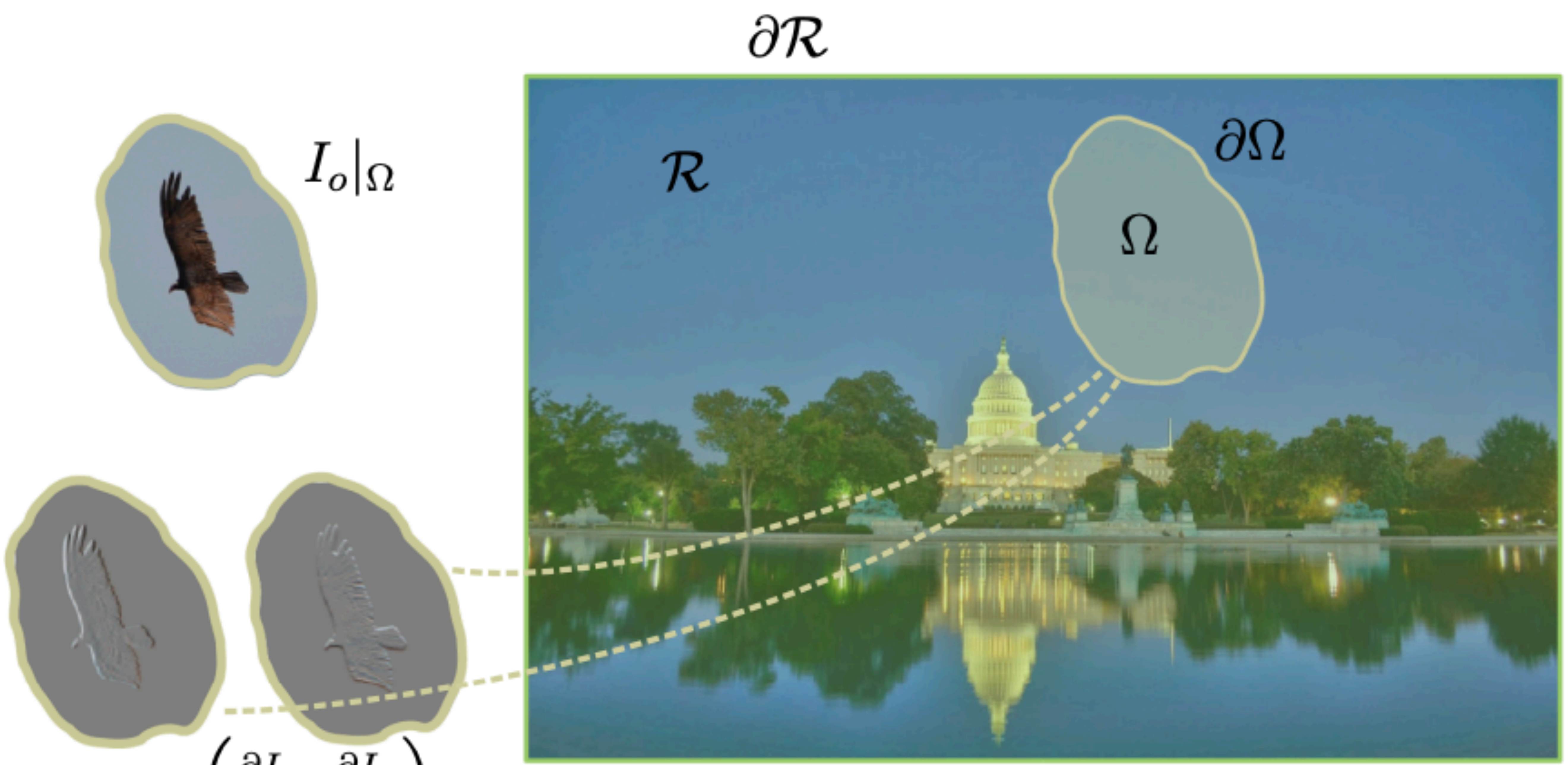
Keywords: Poisson editing; image gradient; integration; Poisson equation; seamless cloning; image filtering

1 Introduction

Methods based on the manipulation of image gradients are a powerful tool for processing or combining images. For example operations such as *seamless cloning*, *local illumination changes*, *texture flattening* or *seamless tiling* can be performed in a very simple and efficient way by combining/modifying the image gradients as illustrated in Figure 1. In addition, many practical applications allow to retrieve the gradient field of different physical quantities of interest; for example, Photometric Stereo (PS) [33], Shape from Shading (SfS) [15] and Differential 3D (D3D) [11], retrieve the gradient field

[1] https://doi.org/10.5201/ipol.2016.163

J. Matías Di Martino, Gabriele Facciolo, Enric Meinhardt-Llopis, *Poisson Image Editing*, Image Processing On Line, 6 (2016), pp. 300–325. https://doi.org/10.5201/ipol.2016.163

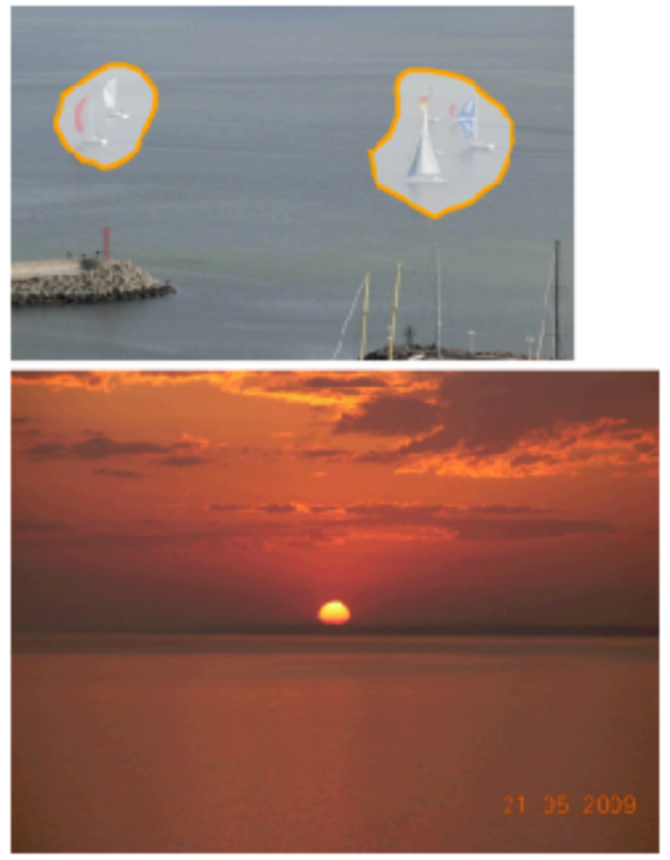


$$\mathbf{v} = \left(\frac{\partial I_o}{\partial x}, \frac{\partial I_o}{\partial y}\right)$$

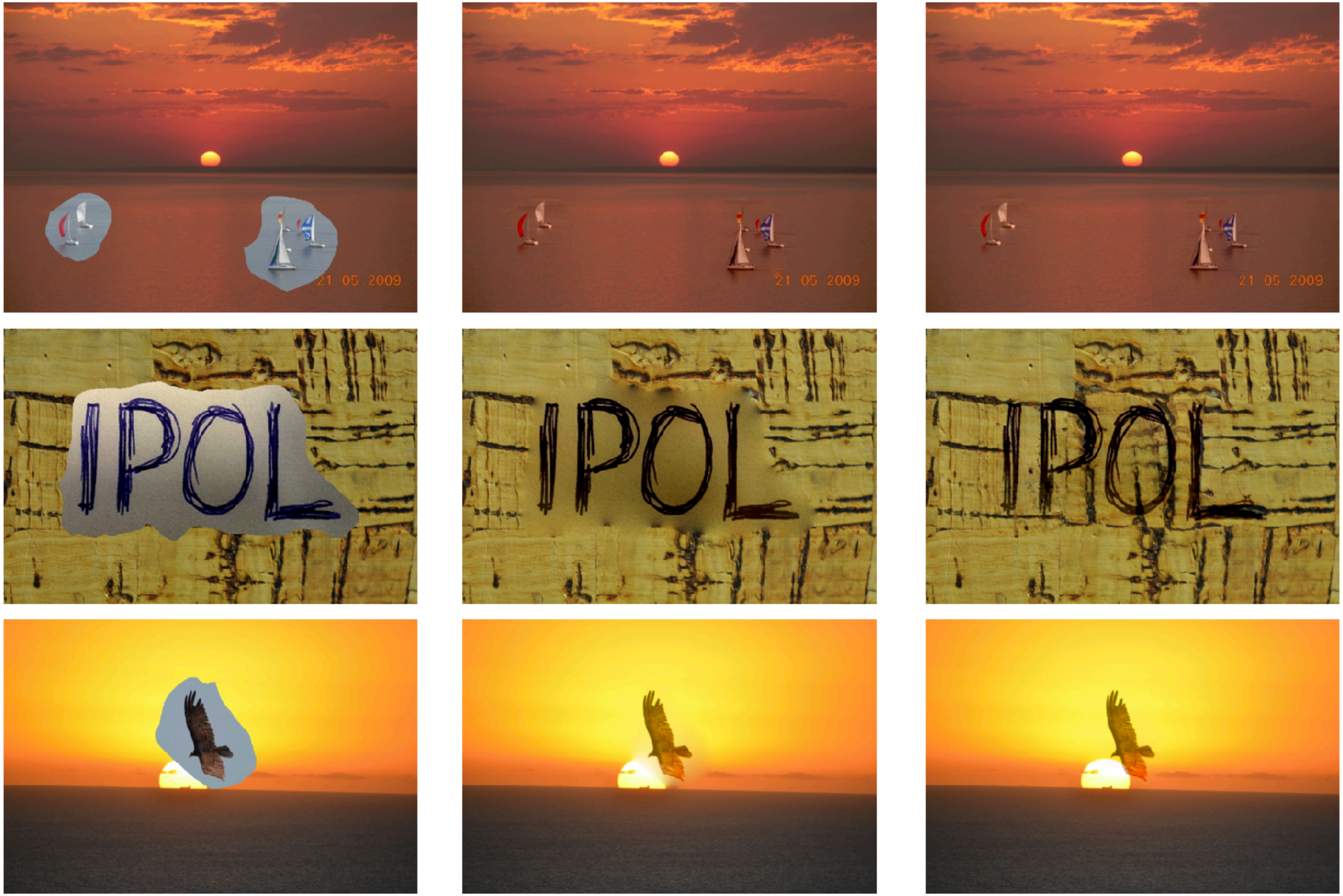
$$\min_{\substack{f \in \mathscr{C}^2(\mathcal{R}) \\ \text{st. } f|_{\mathcal{R}\setminus\Omega} = f^*|_{\mathcal{R}\setminus\Omega}}} \int_\Omega \|\nabla f - \mathbf{v}\|^2 \, dx,$$

$$\Delta f(x) = \operatorname{div}(\mathbf{v}(x)) \quad \text{for all } x \in \Omega, \text{ and } f|_{\partial\Omega} = f^*|_{\partial\Omega},$$

Input

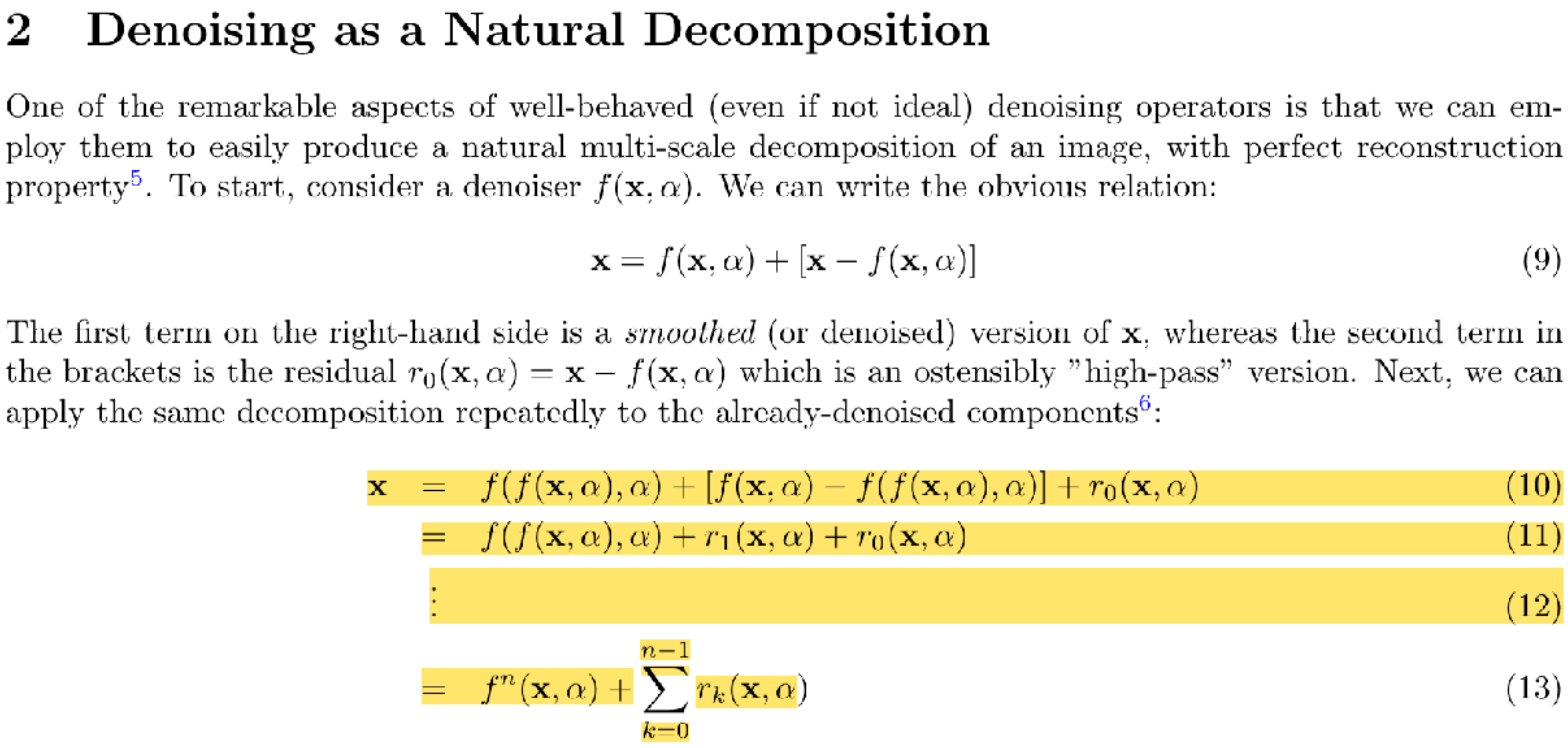


Copy-paste | *Replace* | *Maximum*



$$\min_{\substack{f \,\in\, \mathscr{C}^2(\mathcal{R}) \\ \text{st. } f|_{\mathcal{R}\setminus\Omega} = f^*|_{\mathcal{R}\setminus\Omega}}} \int_\Omega \|\nabla f - \mathbf{v}\|^2 \, dx,$$

Other good thing about denoising: Comes with multi-scale analysis for free



2 Denoising as a Natural Decomposition

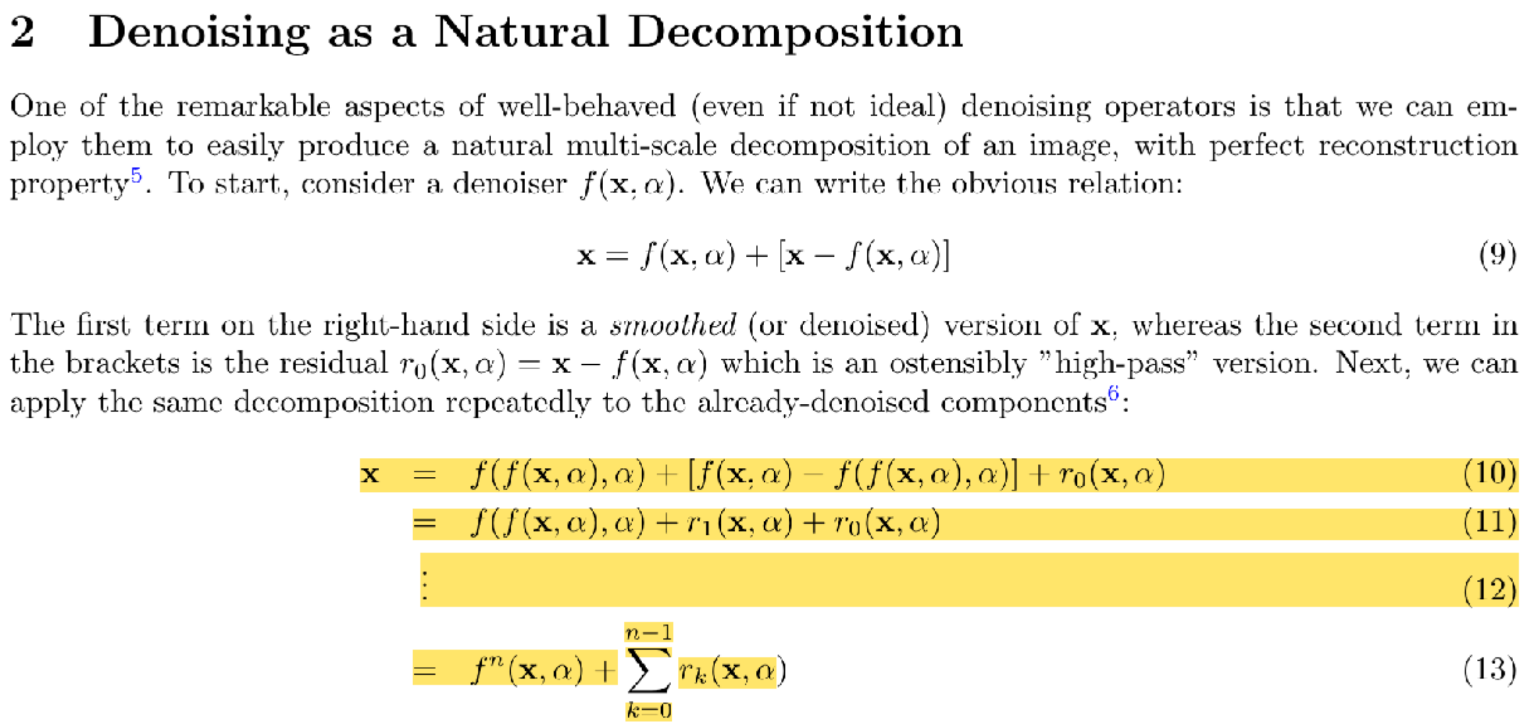
One of the remarkable aspects of well-behaved (even if not ideal) denoising operators is that we can employ them to easily produce a natural multi-scale decomposition of an image, with perfect reconstruction property[5]. To start, consider a denoiser $f(\mathbf{x}, \alpha)$. We can write the obvious relation:

$$\mathbf{x} = f(\mathbf{x}, \alpha) + [\mathbf{x} - f(\mathbf{x}, \alpha)] \tag{9}$$

The first term on the right-hand side is a *smoothed* (or denoised) version of $\mathbf{x}$, whereas the second term in the brackets is the residual $r_0(\mathbf{x}, \alpha) = \mathbf{x} - f(\mathbf{x}, \alpha)$ which is an ostensibly "high-pass" version. Next, we can apply the same decomposition repeatedly to the already-denoised components[6]:

$$\begin{aligned}
\mathbf{x} &= f(f(\mathbf{x}, \alpha), \alpha) + [f(\mathbf{x}, \alpha) - f(f(\mathbf{x}, \alpha), \alpha)] + r_0(\mathbf{x}, \alpha) && (10)\\
&= f(f(\mathbf{x}, \alpha), \alpha) + r_1(\mathbf{x}, \alpha) + r_0(\mathbf{x}, \alpha) && (11)\\
&\vdots && (12)\\
&= f^n(\mathbf{x}, \alpha) + \sum_{k=0}^{n-1} r_k(\mathbf{x}, \alpha) && (13)
\end{aligned}$$

Milanfar, P. & Delbracio, M. Denoising: A Powerful Building-Block for Imaging, Inverse Problems, and Machine Learning. Preprint at http://arxiv.org/abs/2409.06219 (2024).



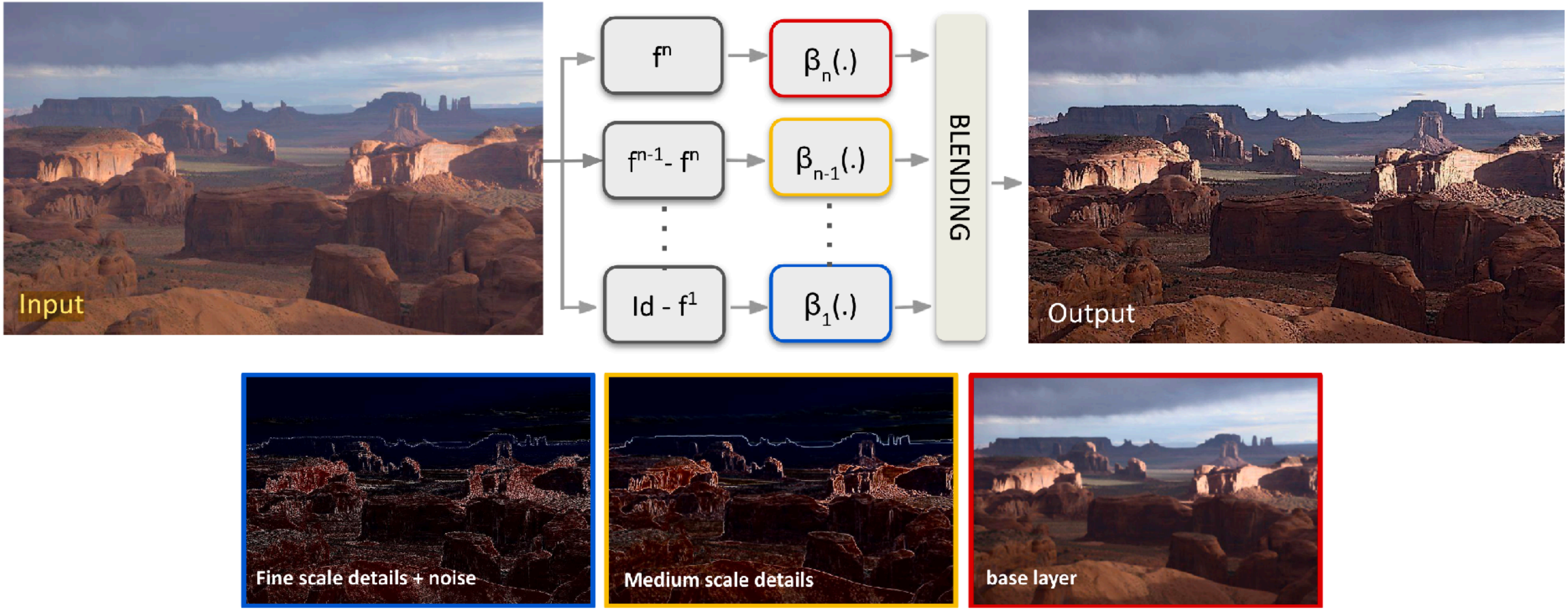
2 Denoising as a Natural Decomposition

One of the remarkable aspects of well-behaved (even if not ideal) denoising operators is that we can employ them to easily produce a natural multi-scale decomposition of an image, with perfect reconstruction property[5]. To start, consider a denoiser $f(\mathbf{x}, \alpha)$. We can write the obvious relation:

$$\mathbf{x} = f(\mathbf{x}, \alpha) + [\mathbf{x} - f(\mathbf{x}, \alpha)] \tag{9}$$

The first term on the right-hand side is a *smoothed* (or denoised) version of $\mathbf{x}$, whereas the second term in the brackets is the residual $r_0(\mathbf{x}, \alpha) = \mathbf{x} - f(\mathbf{x}, \alpha)$ which is an ostensibly "high-pass" version. Next, we can apply the same decomposition repeatedly to the already-denoised components[6]:

$$\begin{aligned} \mathbf{x} &= f(f(\mathbf{x}, \alpha), \alpha) + [f(\mathbf{x}, \alpha) - f(f(\mathbf{x}, \alpha), \alpha)] + r_0(\mathbf{x}, \alpha) && (10) \\ &= f(f(\mathbf{x}, \alpha), \alpha) + r_1(\mathbf{x}, \alpha) + r_0(\mathbf{x}, \alpha) && (11) \\ &\vdots && (12) \\ &= f^n(\mathbf{x}, \alpha) + \sum_{k=0}^{n-1} r_k(\mathbf{x}, \alpha) && (13) \end{aligned}$$



Milanfar, P. & Delbracio, M. Denoising: A Powerful Building-Block for Imaging, Inverse Problems, and Machine Learning. Preprint at http://arxiv.org/abs/2409.06219 (2024).

Give me my money back; I was promised AI

$$\hat{x} = \text{argmin}_x f(x) =$$

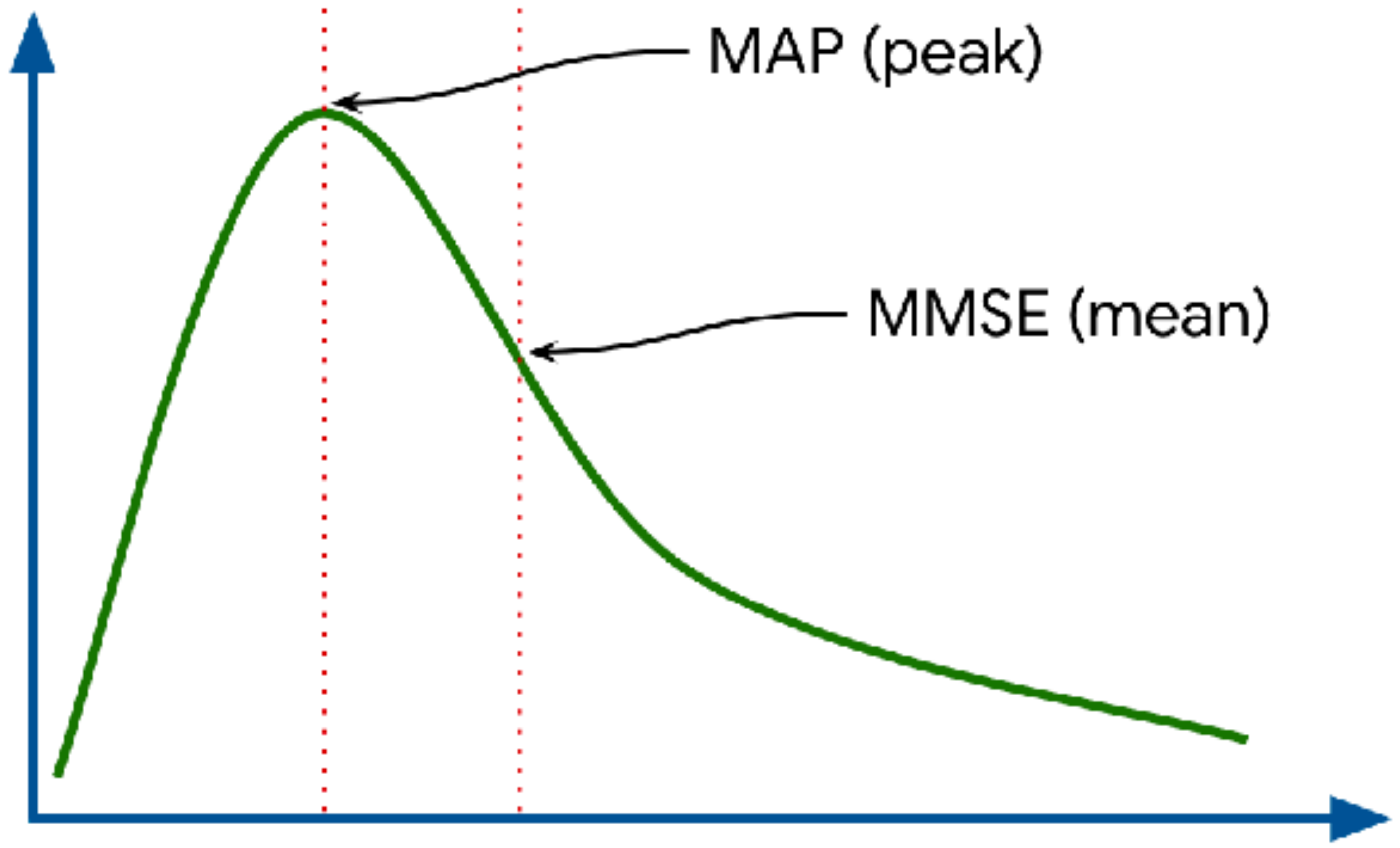
$$f(x) = \underbrace{\|x - y\|_2^2}_{\text{Fitting Data}} + \underbrace{R(x)}_{\text{Regularization}}$$

$$\hat{x} = \text{argmax}_x P(x\,|\,y) \rightarrow \quad \hat{x} = \text{argmin}_x - \log p(y\,|\,x) - \log p(x)$$

Recall, Bayes Theorem:

$$P(x\,|\,y) = \frac{P(y\,|\,x)P(x)}{P(y)} \rightarrow \text{argmax}\, P(x\,|\,y) = \text{argmax}\, P(y\,|\,x)P(x)$$

$$P(y\,|\,x) = P(n) = \frac{1}{\sigma\sqrt{2\pi}}e^{-\frac{1}{2}\frac{n^2}{\sigma^2}} = \frac{1}{\sigma\sqrt{2\pi}}e^{-\frac{1}{2}\frac{(x-y)^2}{\sigma^2}} \rightarrow -log(P(y\,|\,x)) = C(x-y)^2$$



$$f(x) = \|x - y\|_2^2 + R(x)$$

Fitting Data Regularization

Energy

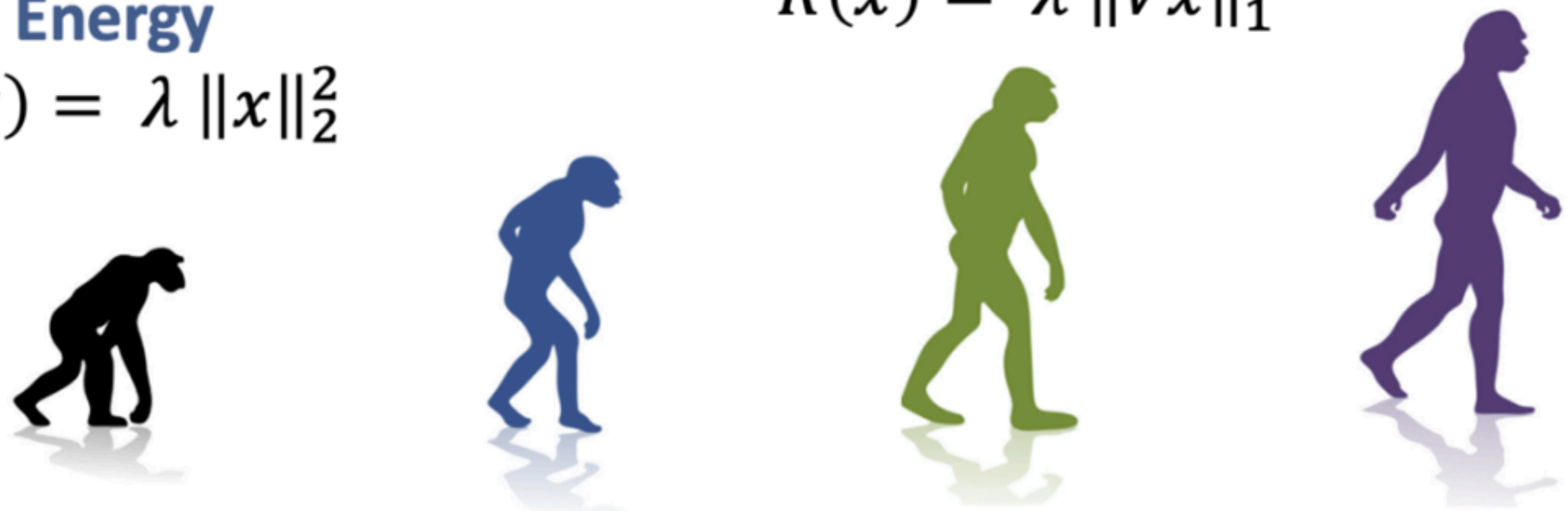
$$R(x) = \lambda \, \|x\|_2^2$$

Smoothness

$$R(x) = \lambda \, \|\boldsymbol{L}x\|_2^2$$

$$f(x) = \|x - y\|_2^2 + R(x)$$

Fitting Data Regularization



Energy

$$R(x) = \lambda \|x\|_2^2$$

Smoothness

$$R(x) = \lambda \|\boldsymbol{L}x\|_2^2$$

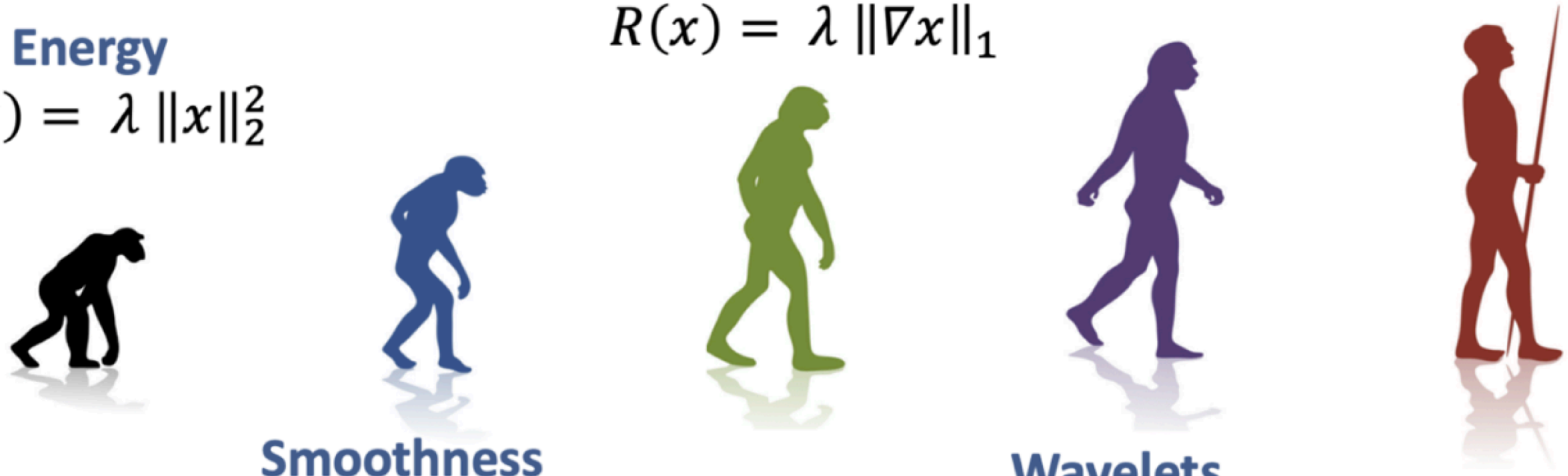
Total Variation

$$R(x) = \lambda \|\nabla x\|_1$$

Wavelets

$$R(x) = \lambda \|\mathrm{W}x\|_1$$

$$f(x) = \underbrace{\|x - y\|_2^2}_{\text{Fitting Data}} + \underbrace{R(x)}_{\text{Regularization}}$$



Energy

$R(x) = \lambda \|x\|_2^2$

Smoothness

$R(x) = \lambda \|\boldsymbol{L}x\|_2^2$

Total Variation

$R(x) = \lambda \|\nabla x\|_1$

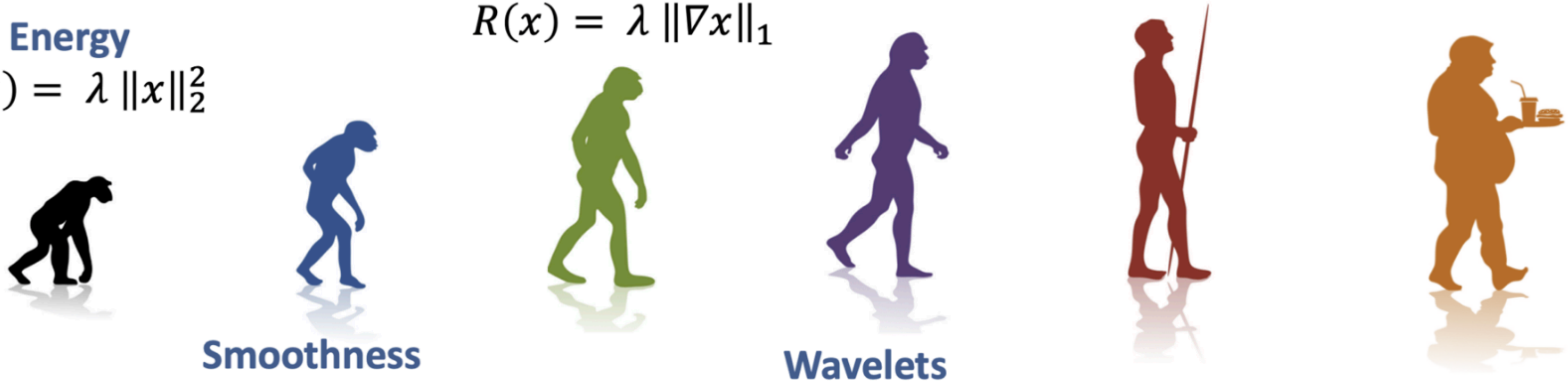
Wavelets

$R(x) = \lambda \|\mathrm{W}x\|_1$

Sparse & Redundant Representations

$R(x) = \lambda \|\gamma\|_0$ for $\boldsymbol{D}\gamma = x$

$$f(x) = \underbrace{\|x - y\|_2^2}_{\text{Fitting Data}} + \underbrace{R(x)}_{\text{Regularization}}$$



Energy

$R(x) = \lambda \|x\|_2^2$

Smoothness

$R(x) = \lambda \|\boldsymbol{L}x\|_2^2$

Total Variation

$R(x) = \lambda \|\nabla x\|_1$

Wavelets

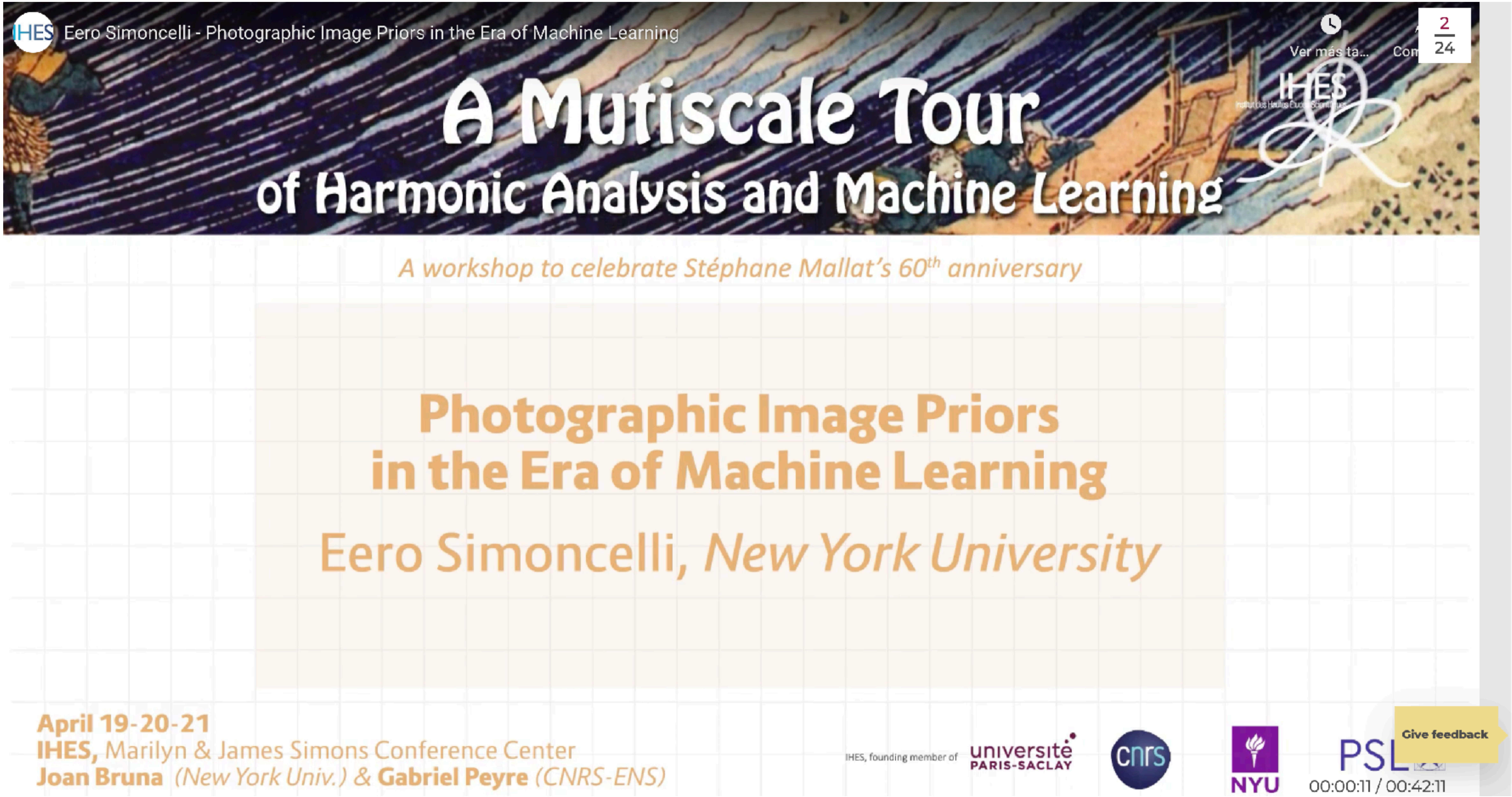
$R(x) = \lambda \|Wx\|_1$

Sparse & Redundant Representations

$R(x) = \lambda \|\gamma\|_0$ for $\boldsymbol{D}\gamma = x$

Deep Learning

Image credit: Jeremi Sulam



IHES
Eero Simoncelli - Photographic Image Priors in the Era of Machine Learning
Ver más ta...
Com
2
24
IHES
Give feedback
00:00:11 / 00:42:11

A Mutiscale Tour of Harmonic Analysis and Machine Learning

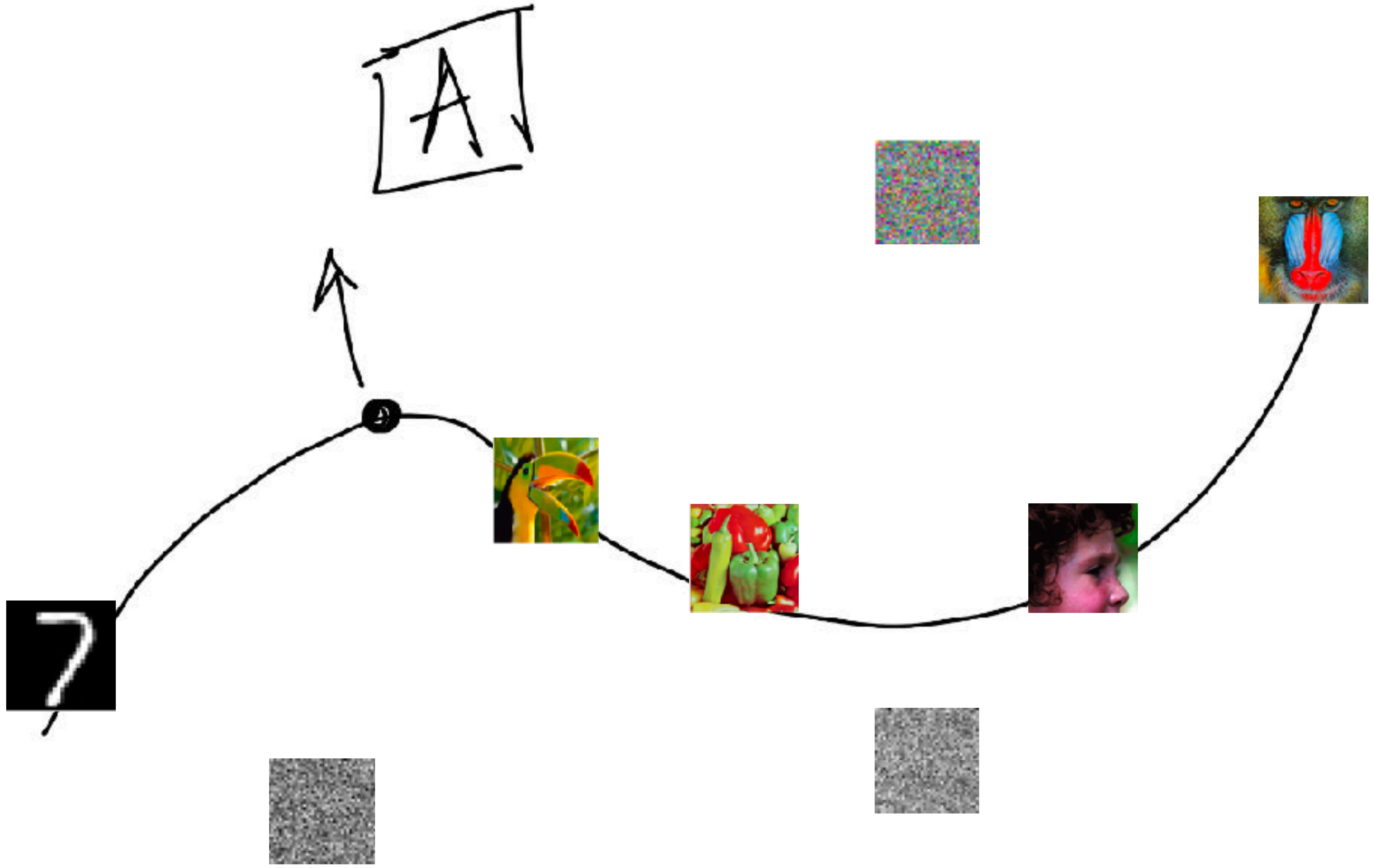
A workshop to celebrate Stéphane Mallat's 60th anniversary

Photographic Image Priors in the Era of Machine Learning

Eero Simoncelli, *New York University*

April 19-20-21
IHES, Marilyn & James Simons Conference Center
Joan Bruna *(New York Univ.)* & **Gabriel Peyre** *(CNRS-ENS)*

https://www.carmin.tv/en/collections/a-multiscale-tour-of-harmonic-analysis-and-machine-learning-to-celebrate-stephane-mallats-60th-birth/video/photographic-image-priors-in-the-era-of-machine-learning



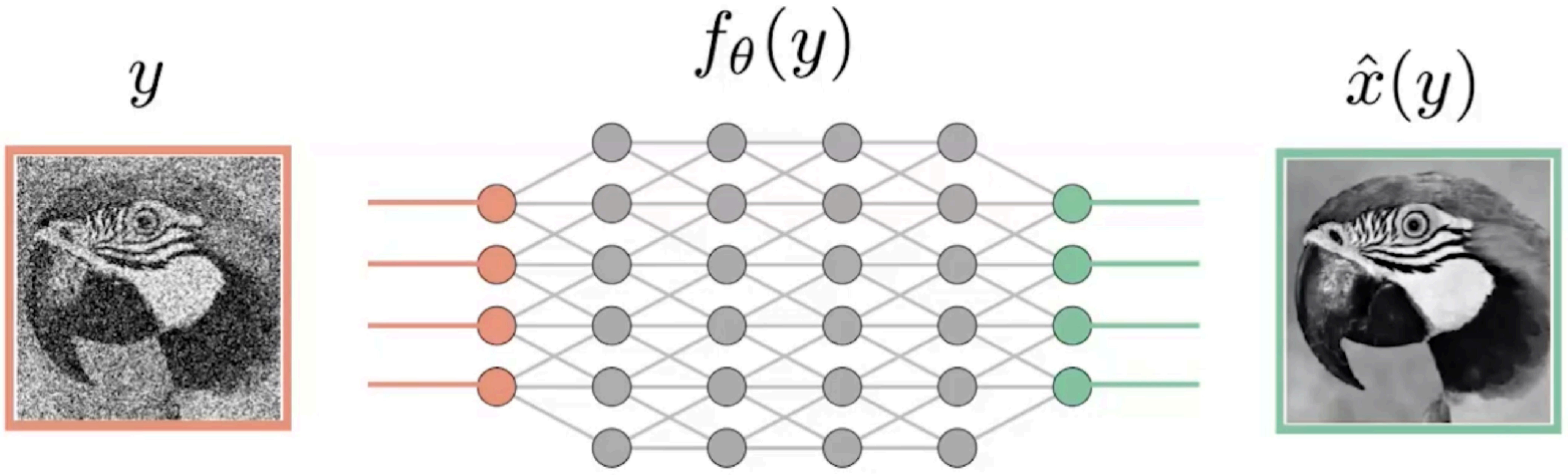
This is a high dimensional space, e.g., image 1000x1000

leaves in a 10^6 dimensional space

https://www.carmin.tv/en/collections/a-multiscale-tour-of-harmonic-analysis-and-machine-learning-to-celebrate-stephane-mallats-60th-birth/video/photographic-image-priors-in-the-era-of-machine-learning

Kadkhodaie, Z., & Simoncelli, E. P. (2021). Solving Linear Inverse Problems Using the Prior Implicit in a Denoiser (arXiv:2007.13640). arXiv. http://arxiv.org/abs/2007.13640

Denoising with CNNs



This thing has an implicit prior p(x), how we pull it out?

https://www.carmin.tv/en/collections/a-multiscale-tour-of-harmonic-analysis-and-machine-learning-to-celebrate-stephane-mallats-60th-birth/video/photographic-image-priors-in-the-era-of-machine-learning

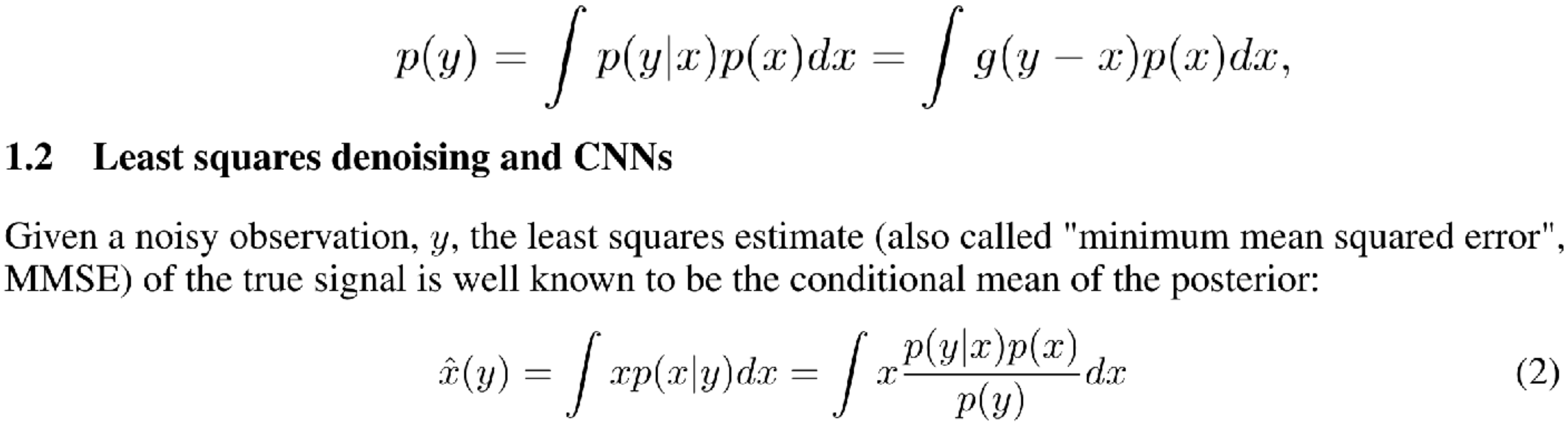
Kadkhodaie, Z., & Simoncelli, E. P. (2021). Solving Linear Inverse Problems Using the Prior Implicit in a Denoiser (arXiv:2007.13640). arXiv. http://arxiv.org/abs/2007.13640

Suppose we make a noisy observation of an image, $y = x + z$, where $x \in R^N$ is the original image drawn from $p(x)$, and $z \sim \mathcal{N}(0, \sigma^2 I_N)$ is a sample of Gaussian white noise. The observation density $p(y)$ (also known as the *prior predictive density*) is related to the prior $p(x)$ via marginalization:

$$p(y) = \int p(y|x)p(x)dx = \int g(y - x)p(x)dx, \tag{1}$$

where the noise distribution is

$$g(z) = \frac{1}{(2\pi\sigma^2)^{N/2}} e^{-||z||^2/2\sigma^2}.$$



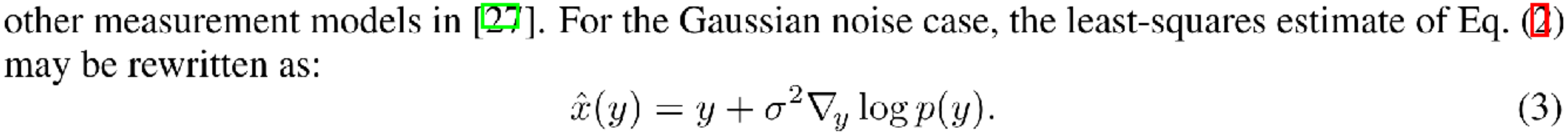
$$p(y) = \int p(y|x)p(x)dx = \int g(y-x)p(x)dx,$$

1.2 Least squares denoising and CNNs

Given a noisy observation, y, the least squares estimate (also called "minimum mean squared error", MMSE) of the true signal is well known to be the conditional mean of the posterior:

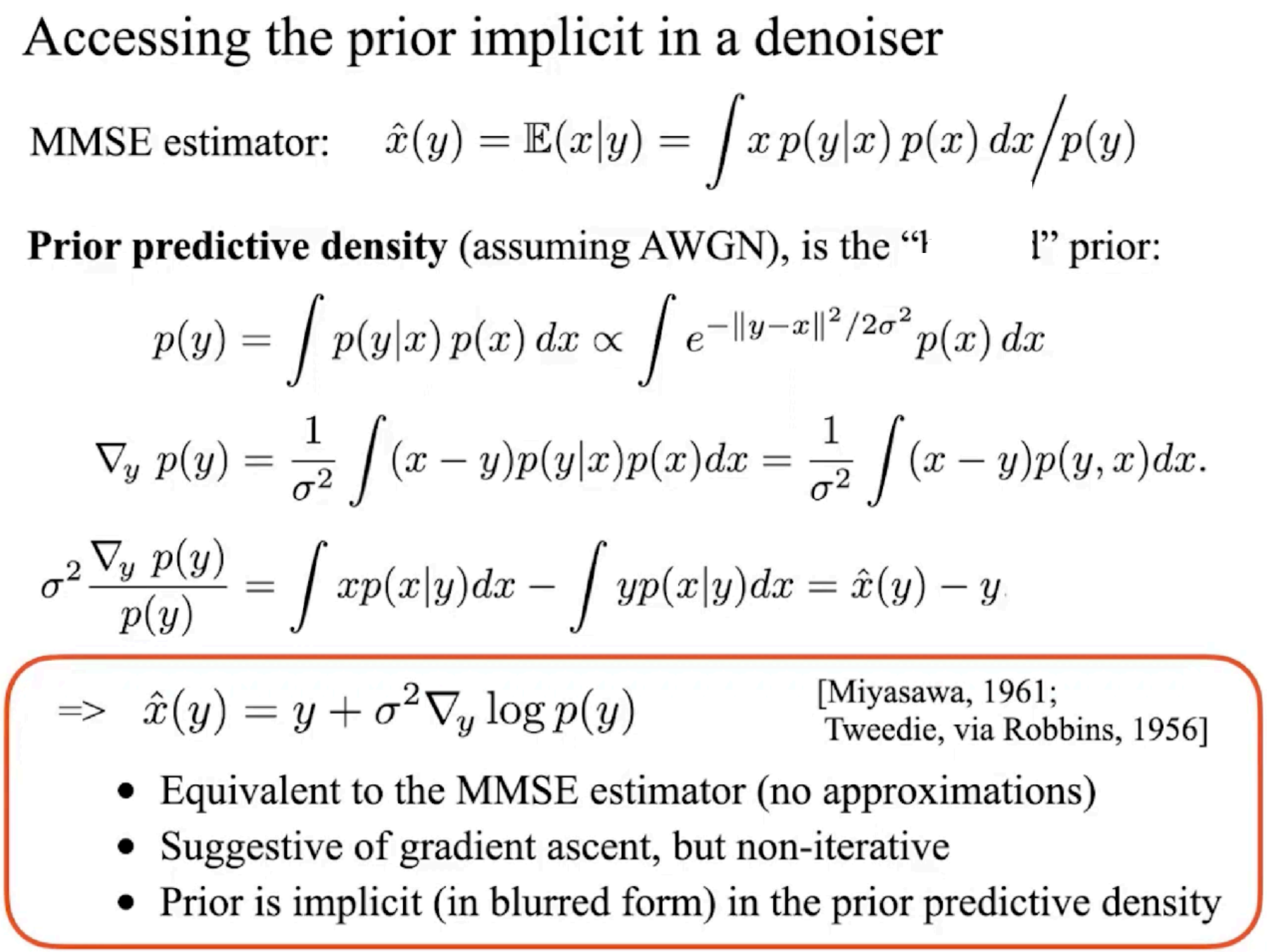
$$\hat{x}(y) = \int xp(x|y)dx = \int x\frac{p(y|x)p(x)}{p(y)}dx \tag{2}$$

For example, minimizing the denoising MSE over a large training set of example signals and their noise-corrupted counterparts



other measurement models in [27]. For the Gaussian noise case, the least-squares estimate of Eq. (2) may be rewritten as:

$$\hat{x}(y) = y + \sigma^2\nabla_y \log p(y). \tag{3}$$



Accessing the prior implicit in a denoiser

MMSE estimator: $\hat{x}(y) = \mathbb{E}(x|y) = \int x\, p(y|x)\, p(x)\, dx \Big/ p(y)$

Prior predictive density (assuming AWGN), is the “b[illegible]d” prior:

$$p(y) = \int p(y|x)\, p(x)\, dx \propto \int e^{-\|y-x\|^2/2\sigma^2} p(x)\, dx$$

$$\nabla_y\, p(y) = \frac{1}{\sigma^2} \int (x-y) p(y|x) p(x) dx = \frac{1}{\sigma^2} \int (x-y) p(y,x) dx.$$

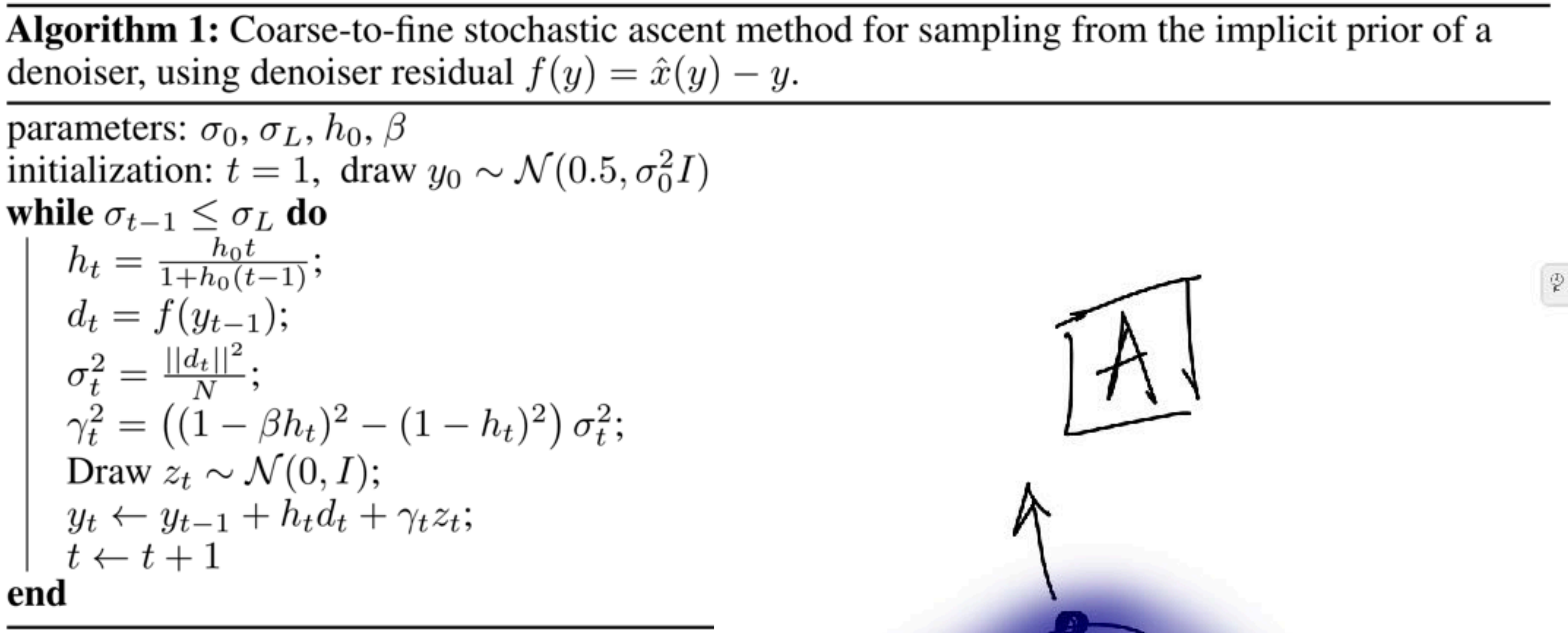
$$\sigma^2 \frac{\nabla_y\, p(y)}{p(y)} = \int x p(x|y) dx - \int y p(x|y) dx = \hat{x}(y) - y$$

=> $\hat{x}(y) = y + \sigma^2 \nabla_y \log p(y)$

[Miyasawa, 1961;
Tweedie, via Robbins, 1956]

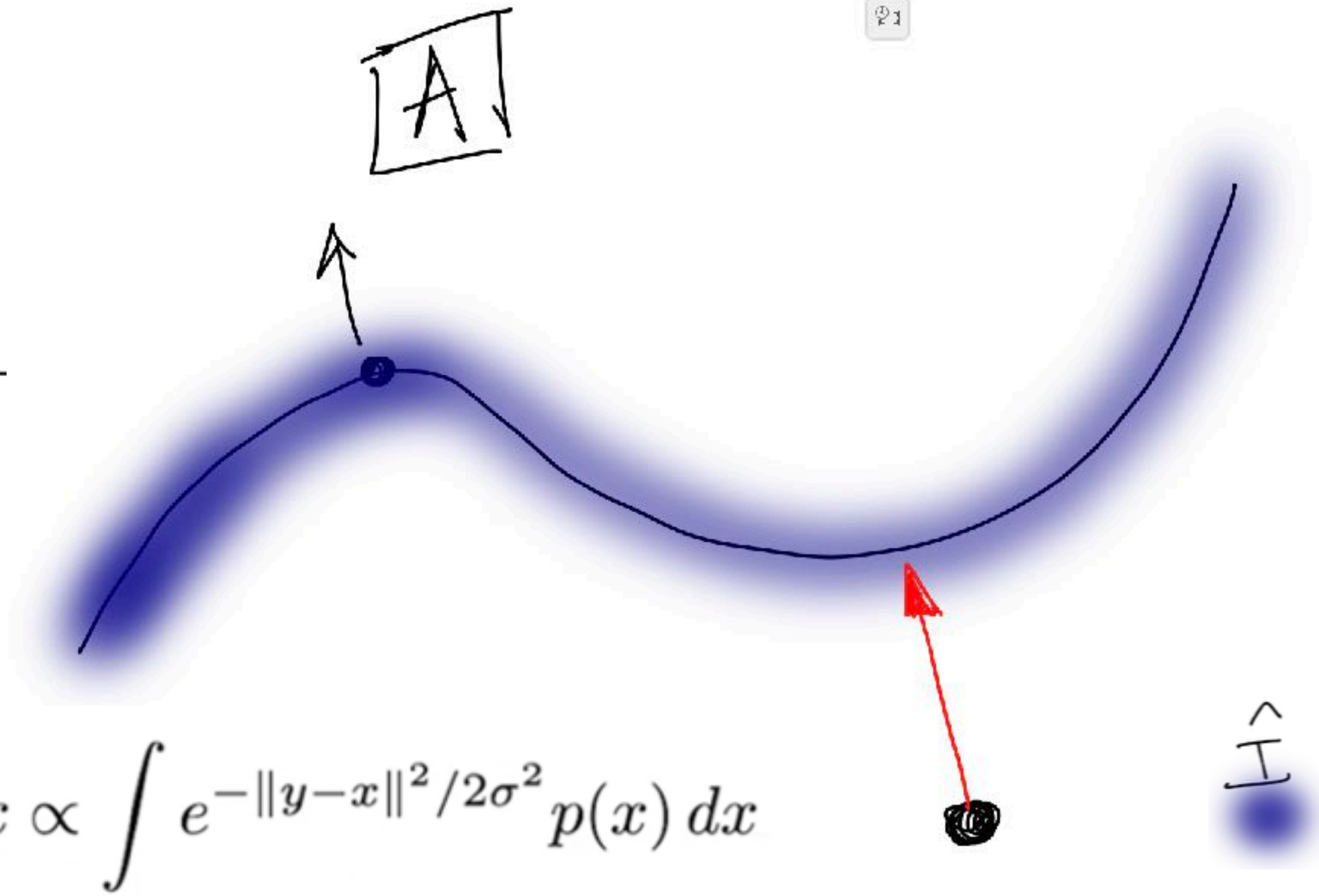
- Equivalent to the MMSE estimator (no approximations)
- Suggestive of gradient ascent, but non-iterative
- Prior is implicit (in blurred form) in the prior predictive density

Kadkhodaie, Zahra, and Eero Simoncelli. "Stochastic solutions for linear inverse problems using the prior implicit in a denoiser." *Advances in Neural Information Processing Systems* 34 (2021): 13242-13254.



Algorithm 1: Coarse-to-fine stochastic ascent method for sampling from the implicit prior of a denoiser, using denoiser residual $f(y) = \hat{x}(y) - y$.

```
parameters: $\sigma_0, \sigma_L, h_0, \beta$
initialization: $t = 1$, draw $y_0 \sim \mathcal{N}(0.5, \sigma_0^2 I)$
while $\sigma_{t-1} \leq \sigma_L$ do
    $h_t = \frac{h_0 t}{1 + h_0(t-1)}$;
    $d_t = f(y_{t-1})$;
    $\sigma_t^2 = \frac{||d_t||^2}{N}$;
    $\gamma_t^2 = \left((1 - \beta h_t)^2 - (1 - h_t)^2\right) \sigma_t^2$;
    Draw $z_t \sim \mathcal{N}(0, I)$;
    $y_t \leftarrow y_{t-1} + h_t d_t + \gamma_t z_t$;
    $t \leftarrow t + 1$
end
```



$$p(y) = \int p(y|x)\, p(x)\, dx \propto \int e^{-\|y-x\|^2/2\sigma^2} p(x)\, dx$$

https://www.carmin.tv/en/collections/a-multiscale-tour-of-harmonic-analysis-and-machine-learning-to-celebrate-stephane-mallats-60th-birth/video/photographic-image-priors-in-the-era-of-machine-learning

Kadkhodaie, Z., & Simoncelli, E. P. (2021). Solving Linear Inverse Problems Using the Prior Implicit in a Denoiser (arXiv:2007.13640). arXiv. http://arxiv.org/abs/2007.13640

C Visualization of Universal Inverse Sampler on a 2D manifold prior

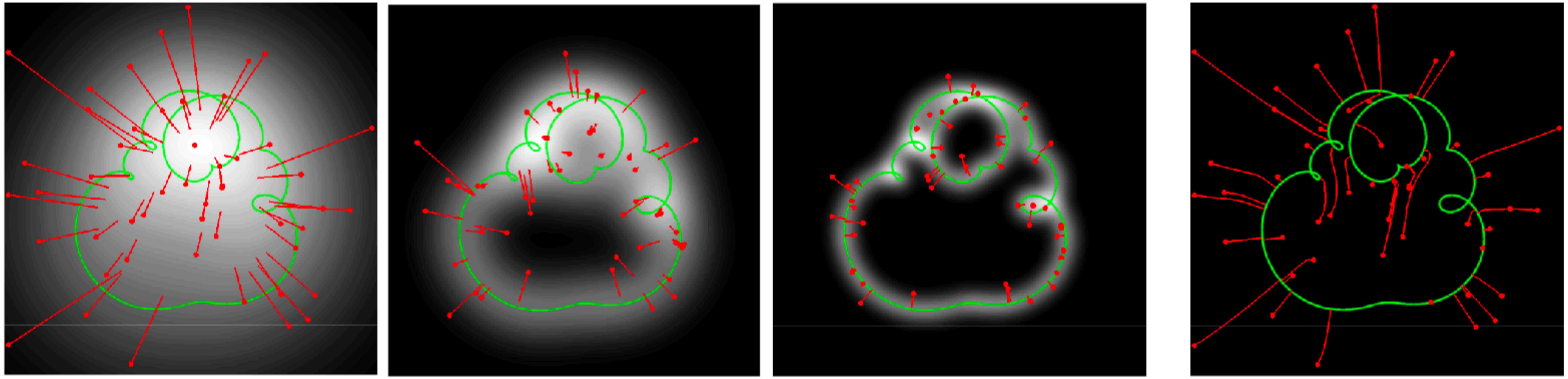


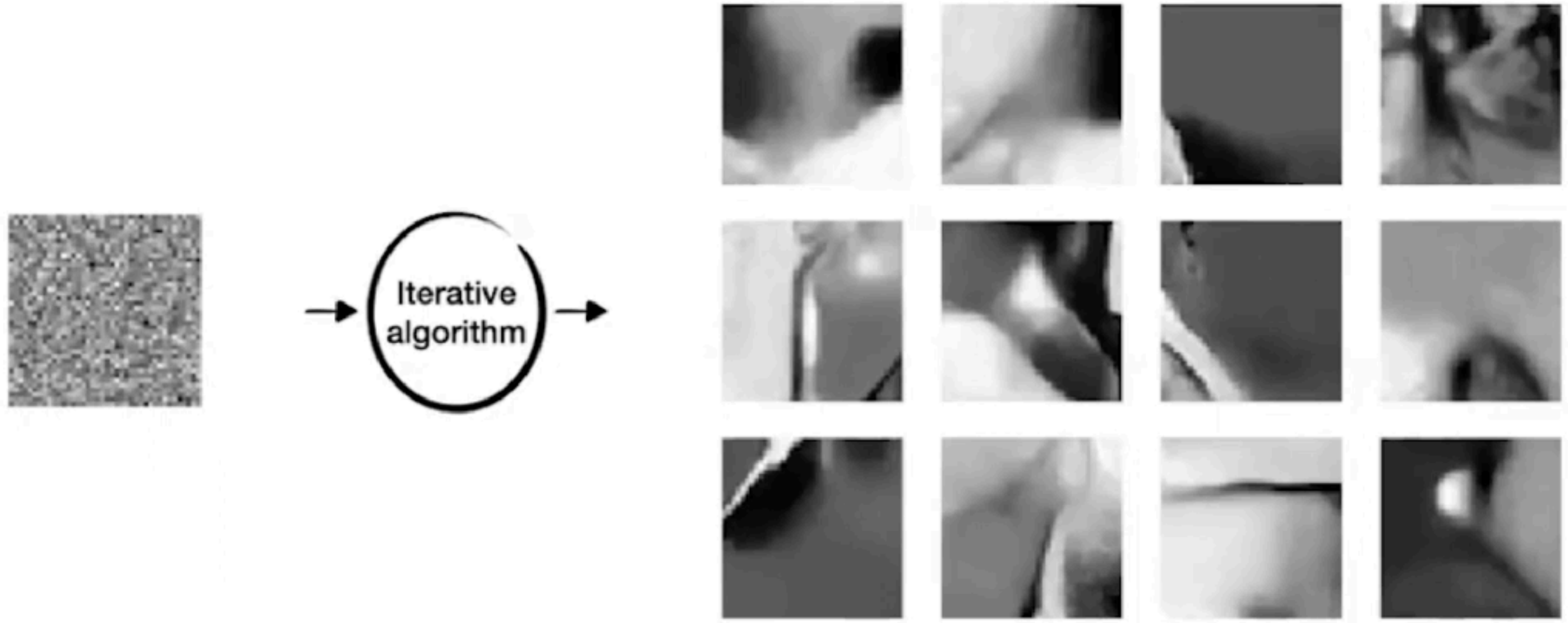
Figure 10: Two-dimensional simulation/visualization of the Universal Inverse Sampler. Fifty example signals x are sampled from a uniform prior on a manifold (green curve). First three panels show, for three different levels of noise, the noise-corrupted measurements of the signals (red points), the associated noisy signal distribution $p(y)$ (indicated with underlying grayscale intensities), and the least-squares optimal denoising solution $\hat{x}(y)$ for each (end of red line segments), as defined by Eq. (2), or equivalently, Eq. (3). Right panel shows trajectory of our iterative coarse-to-fine inverse algorithm (Algorithm 2, depicted in Figure 9), starting from the same initial values y (red points) of the first panel. Algorithm parameters were $h_0 = 0.05$ and $\beta = 1$ (i.e., no injected noise). Note that, unlike the least-squares solutions, the iterative trajectories are curved, and always arrive at solutions on the signal manifold.

Reminder: comment on connections and differences with classical diffusion processes

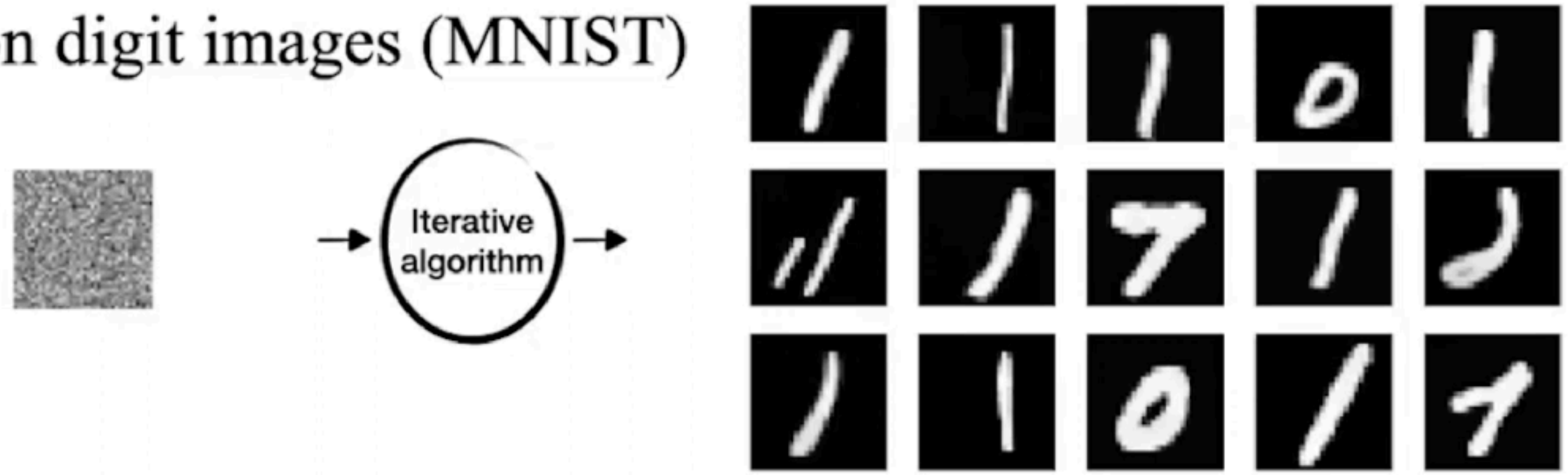
https://www.carmin.tv/en/collections/a-multiscale-tour-of-harmonic-analysis-and-machine-learning-to-celebrate-stephane-mallats-60th-birth/video/photographic-image-priors-in-the-era-of-machine-learning

Kadkhodaie, Z., & Simoncelli, E. P. (2021). Solving Linear Inverse Problems Using the Prior Implicit in a Denoiser (arXiv:2007.13640). arXiv. http://arxiv.org/abs/2007.13640

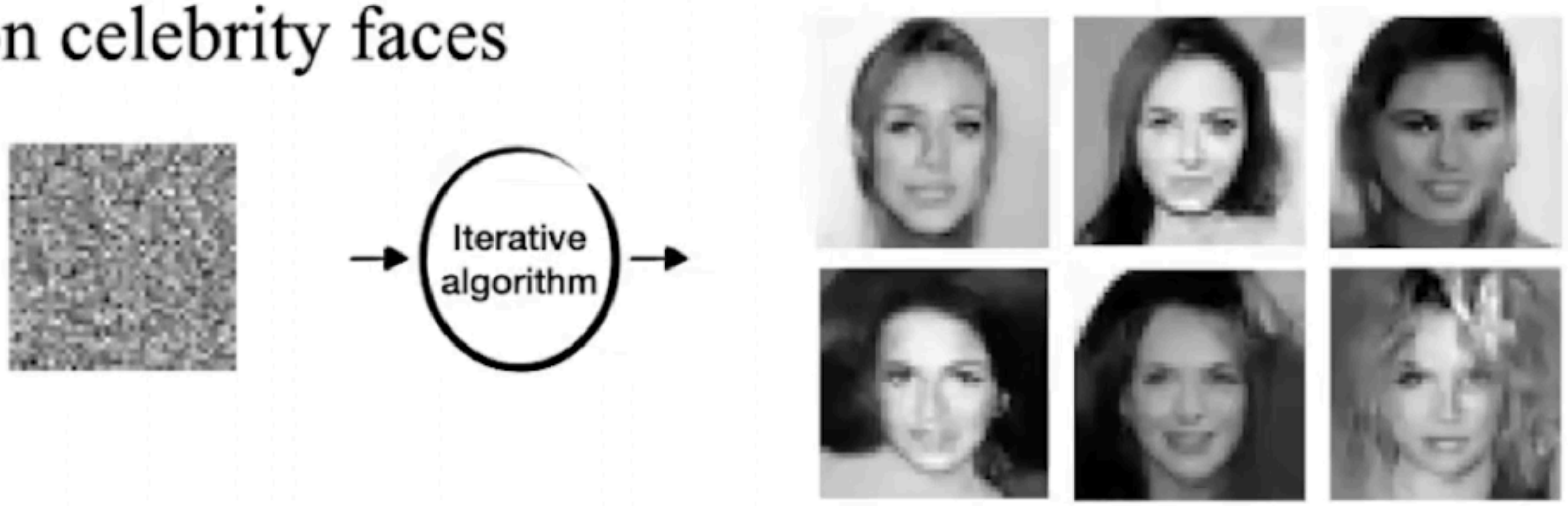
Sampling



Denoiser trained on digit images (MNIST)



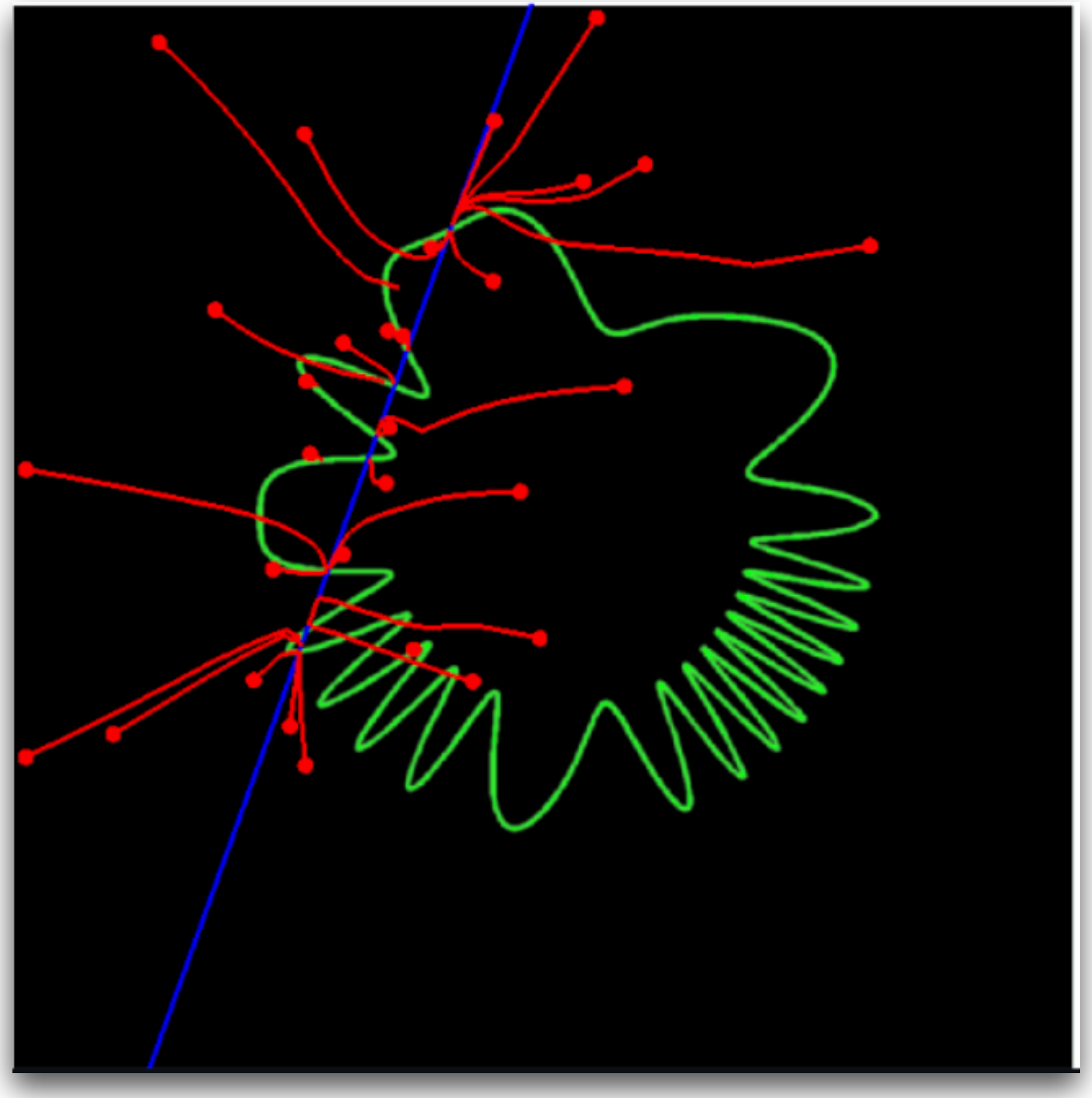
Denoiser trained on celebrity faces



Now, we can use this to solve inverse problems

3 Solving linear inverse problems using the implicit prior

Many applications in signal processing can be expressed as deterministic linear inverse problems - deblurring, super-resolution, estimating missing pixels (e.g., inpainting), and compressive sensing are all examples. Given a set of linear measurements of an image, $x^c = M^T x$, where M is a low-rank measurement matrix, one attempts to recover the original image. In Section 2, we developed a stochastic gradient-ascent algorithm for obtaining a high-probability sample from $p(x)$. Here, we modify this algorithm to solve for a high-probability sample from the conditional density $p(x|M^T x = x^c)$.



3 Solving linear inverse problems using the implicit prior

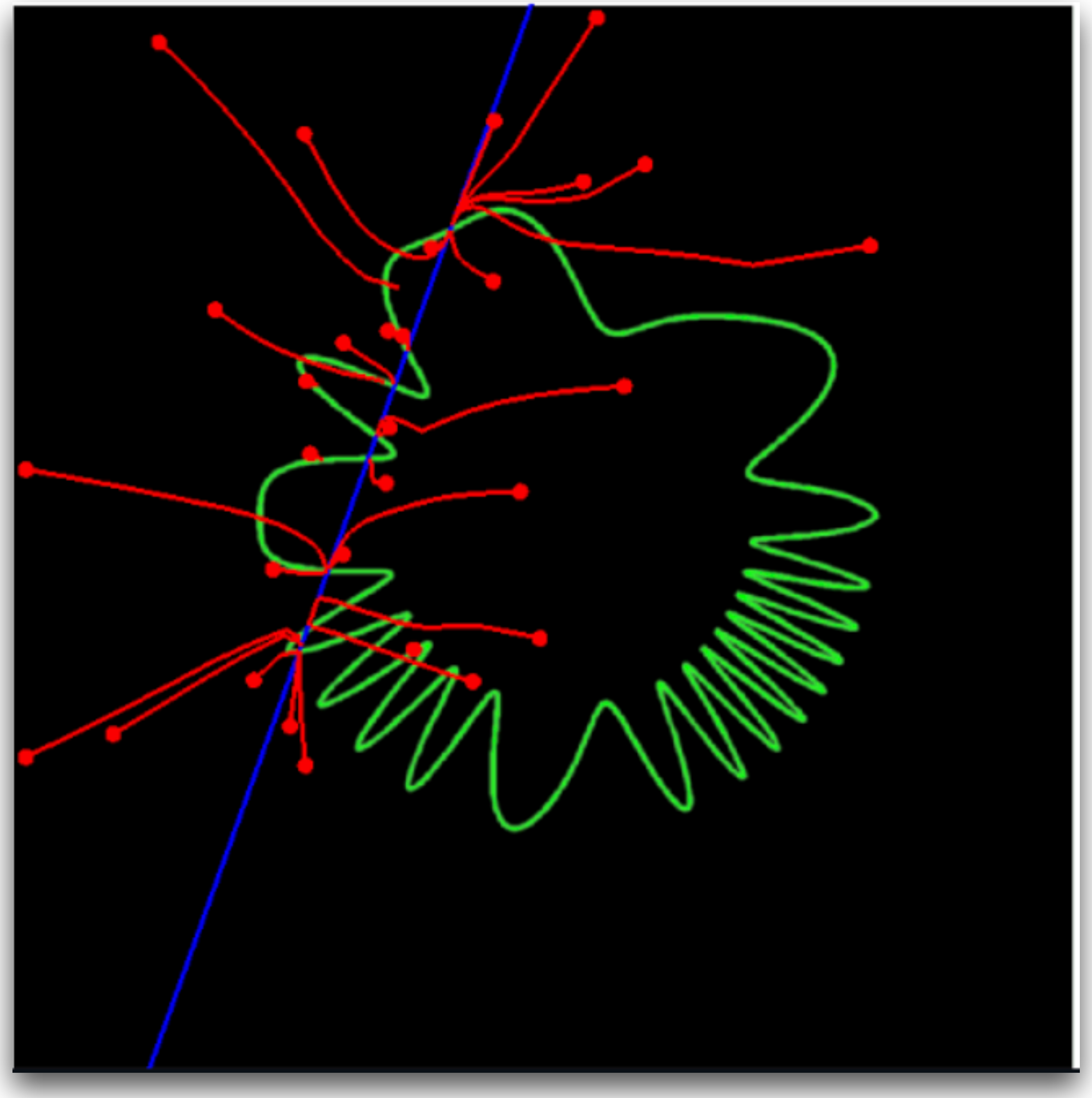
Many applications in signal processing can be expressed as deterministic linear inverse problems - deblurring, super-resolution, estimating missing pixels (e.g., inpainting), and compressive sensing are all examples. Given a set of linear measurements of an image, $x^c = M^T x$, where M is a low-rank measurement matrix, one attempts to recover the original image. In Section 2, we developed a stochastic gradient-ascent algorithm for obtaining a high-probability sample from $p(x)$. Here, we modify this algorithm to solve for a high-probability sample from the conditional density $p(x|M^T x = x^c)$.

$$x = \begin{bmatrix} a & c \\ b & d \end{bmatrix}, \quad x = \begin{pmatrix} a \\ b \\ c \\ d \end{pmatrix}, \quad x^c = \begin{pmatrix} a \\ b \end{pmatrix}$$

$$x^c = M^T x \Rightarrow \begin{pmatrix} a \\ b \end{pmatrix} = \overbrace{\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix}}^{M^T} \begin{pmatrix} a \\ b \\ c \\ d \end{pmatrix}$$

$$M = \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \end{pmatrix} ; \quad MM^T = \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

$$MM^T x = \begin{pmatrix} a & 0 \\ b & 0 \end{pmatrix} \quad \text{(reshaped)}$$

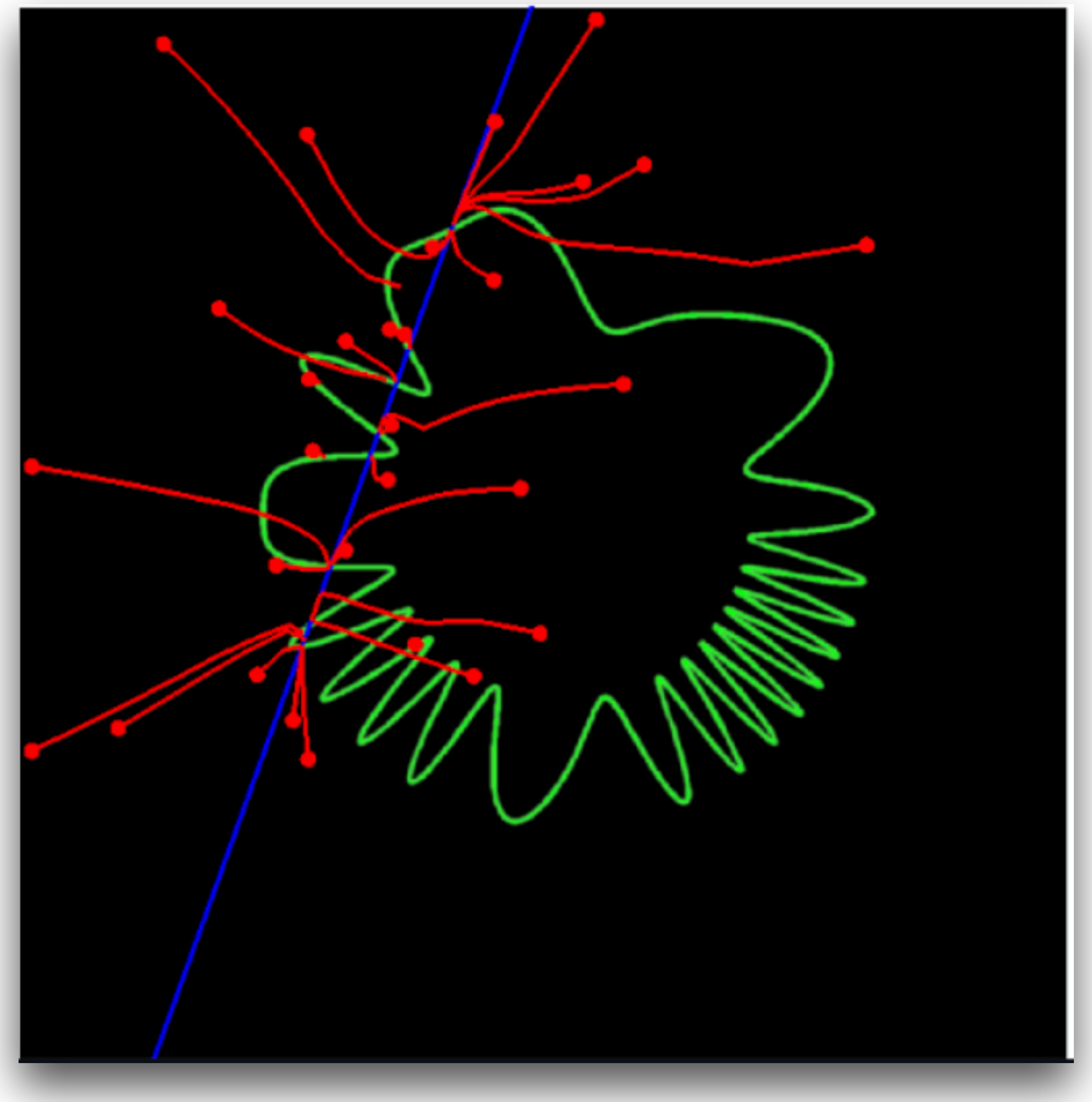


$x = \begin{bmatrix} a & c \\ b & d \end{bmatrix}$, $x = \begin{pmatrix} a \\ b \\ c \\ d \end{pmatrix}$, $x^c = \begin{pmatrix} a \\ b \end{pmatrix}$

$$x^c = M^T x \Rightarrow \begin{pmatrix} a \\ b \end{pmatrix} = \overbrace{\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix}}^{M^T} \begin{pmatrix} a \\ b \\ c \\ d \end{pmatrix}$$

$$M = \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \end{pmatrix} ; \qquad MM^T = \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

$$MM^T x = \begin{pmatrix} a & 0 \\ b & 0 \end{pmatrix} \quad \text{(reshaped)}$$



pseudo-inverse of M^T, and that matrix MM^T can be used to project an image onto the measurement subspace. Using Bayes' rule, we write the conditional density of the noisy image conditioned on the linear measureement as

$$p(y|x^c) = p(y^c, y^u|x^c) = p(y^u|y^c, x^c)p(y^c|x^c) = p(y^u|x^c)p(y^c|x^c)$$

where $y^c = M^T y$, and $y^u = \bar{M}^T y$ (the projection of y onto the orthogonal complement of M). As with the algorithm of Section 2, we wish to obtain a local maximum of this function using stochastic coarse-to-fine gradient ascent. Applying the operator $\sigma^2 \nabla \log(\cdot)$ yields

$$\sigma^2 \nabla_y \log p(y|x^c) = \sigma^2 \nabla_y \log p(y^u|x^c) + \sigma^2 \nabla_y \log p(y^c|x^c).$$

The second term is the gradient of the observation noise distribution, projected into the measurement space. If this is Gaussian with variance σ^2, it reduces to $M(y^c - x^c)$. The first term is the gradient of a function defined only within the subspace orthogonal to the measurements, and thus can be computed by projecting the measurement subspace out of the full gradient. Combining these gives:

$$\begin{aligned} \sigma^2 \nabla_y \log p(y) &= (I - MM^T)\sigma^2 \nabla_y \log p(y) + M(x^c - y^c) \\ &= (I - MM^T)f(y) + M(x^c - M^T y). \end{aligned} \qquad (9)$$

Algorithm 2: Coarse-to-fine stochastic ascent method for sampling on the residual of a denoiser, $f(y) = \hat{x}(y) - y$. Note: e is an imag

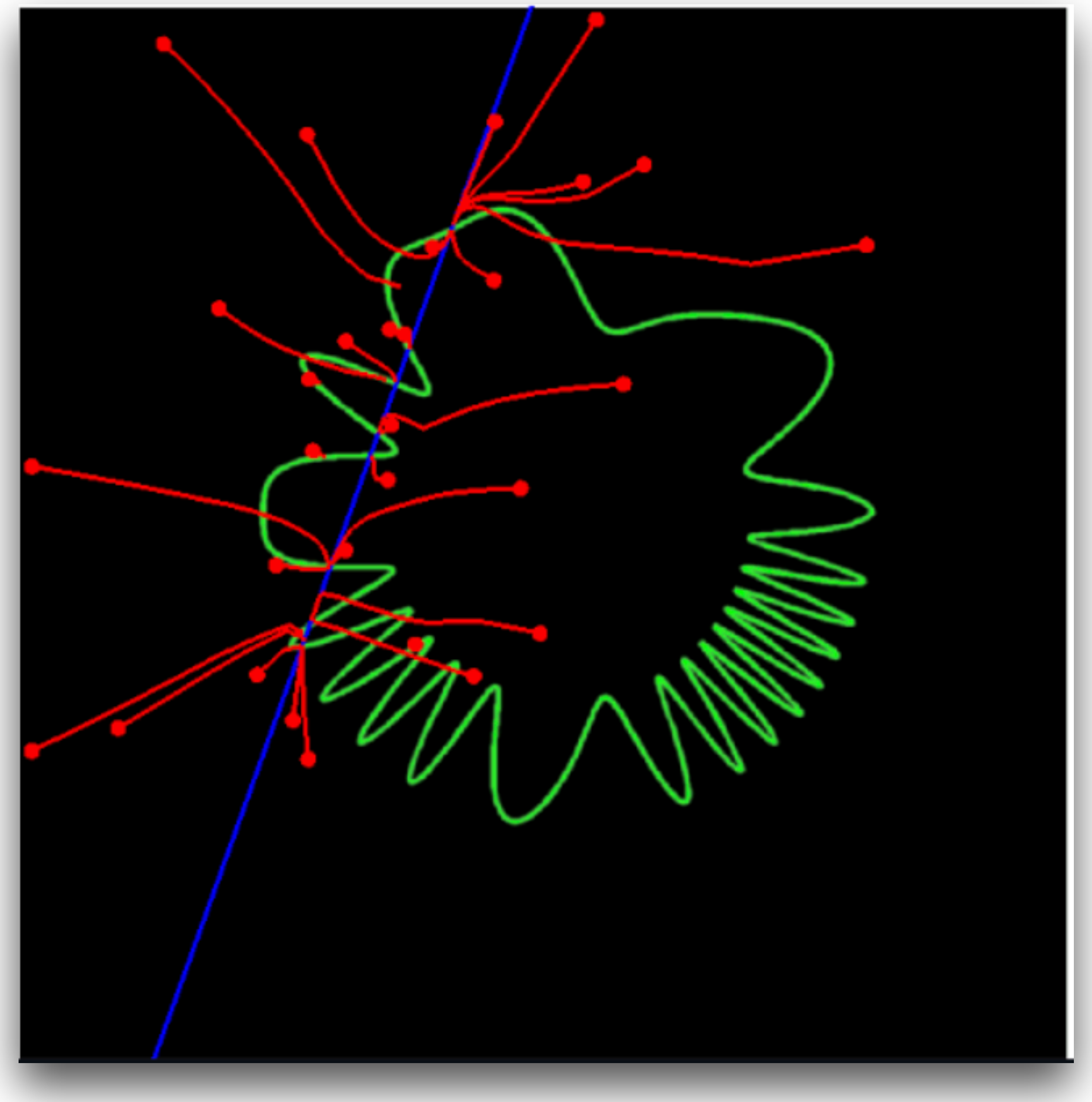
```
parameters: $\sigma_0$, $\sigma_L$, $h_0$, $\beta$, $M$, $x^c$
initialization: t=1; draw $y_0 \sim \mathcal{N}(0.5(I - MM^T)e + Mx^c,\ \sigma_0^2 I)$
while $\sigma_{t-1} \leq \sigma_L$ do
    $h_t = \frac{h_0 t}{1+h_0(t-1)}$;
    $d_t = (I - MM^T)f(y_{t-1}) + M(x^c - M^T y_{t-1})$;
    $\sigma_t^2 = \frac{||d_t||^2}{N}$;
    $\gamma_t^2 = \left((1 - \beta h_t)^2 - (1 - h_t)^2\right)\sigma_t^2$;
    Draw $z_t \sim \mathcal{N}(0, I)$;
    $y_t \leftarrow y_{t-1} + h_t d_t + \gamma_t z_t$;
    $t \leftarrow t + 1$
end
```

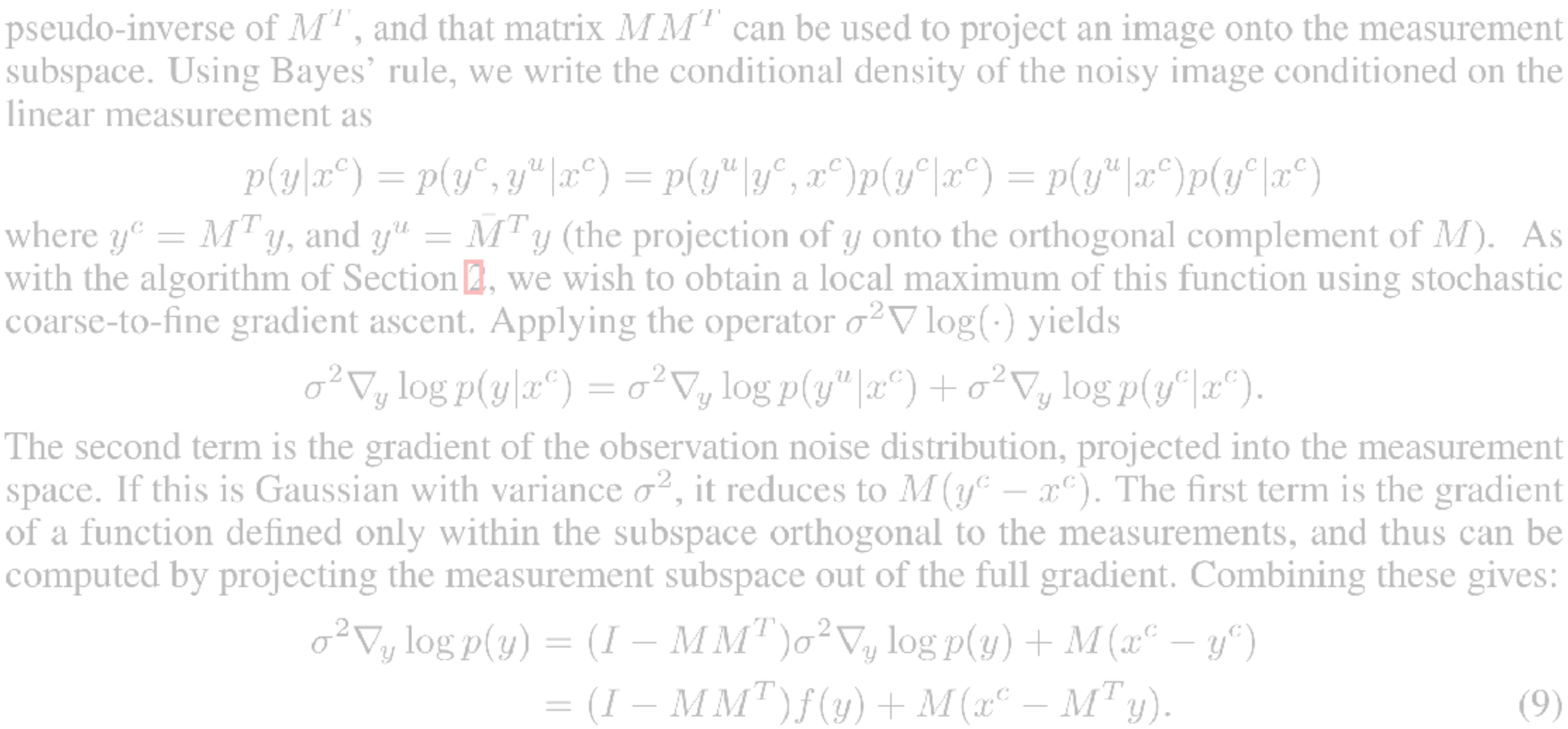
$x = \begin{bmatrix} a & c \\ b & d \end{bmatrix}$, $x = \begin{pmatrix} a \\ b \\ c \\ d \end{pmatrix}$, $x^c = \begin{pmatrix} a \\ b \end{pmatrix}$

$$x^c = M^T x \Rightarrow \begin{pmatrix} a \\ b \end{pmatrix} = \overbrace{\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix}}^{M^T} \begin{pmatrix} a \\ b \\ c \\ d \end{pmatrix}$$

$M = \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \end{pmatrix}$; $MM^T = \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$

$MM^T x = \begin{pmatrix} a & 0 \\ b & 0 \end{pmatrix}$ (reshaped)





pseudo-inverse of M^T, and that matrix MM^T can be used to project an image onto the measurement subspace. Using Bayes' rule, we write the conditional density of the noisy image conditioned on the linear measureement as

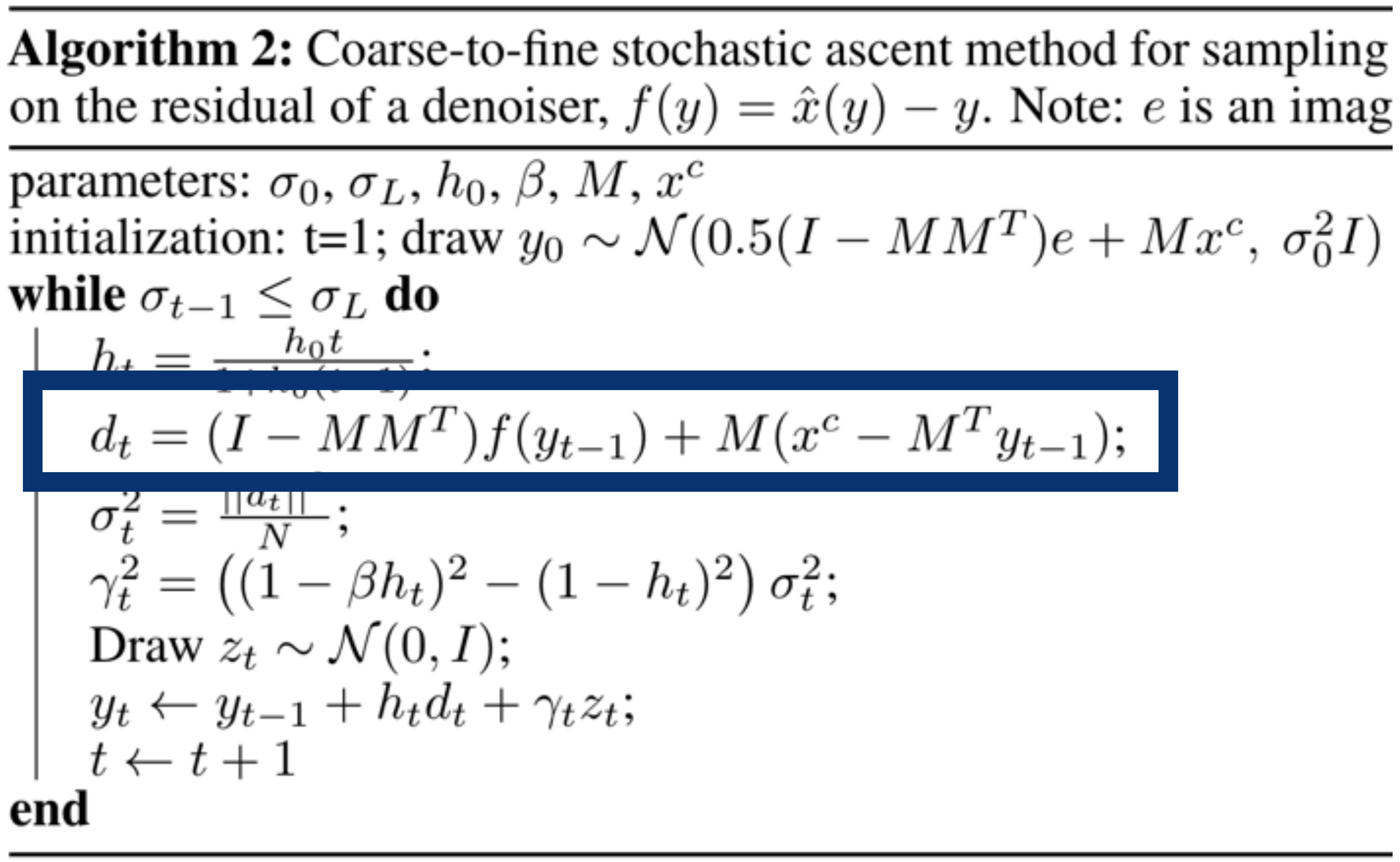
$$p(y|x^c) = p(y^c, y^u|x^c) = p(y^u|y^c, x^c)p(y^c|x^c) = p(y^u|x^c)p(y^c|x^c)$$

where $y^c = M^T y$, and $y^u = \bar{M}^T y$ (the projection of y onto the orthogonal complement of M). As with the algorithm of Section 2, we wish to obtain a local maximum of this function using stochastic coarse-to-fine gradient ascent. Applying the operator $\sigma^2\nabla \log(\cdot)$ yields

$$\sigma^2\nabla_y \log p(y|x^c) = \sigma^2\nabla_y \log p(y^u|x^c) + \sigma^2\nabla_y \log p(y^c|x^c).$$

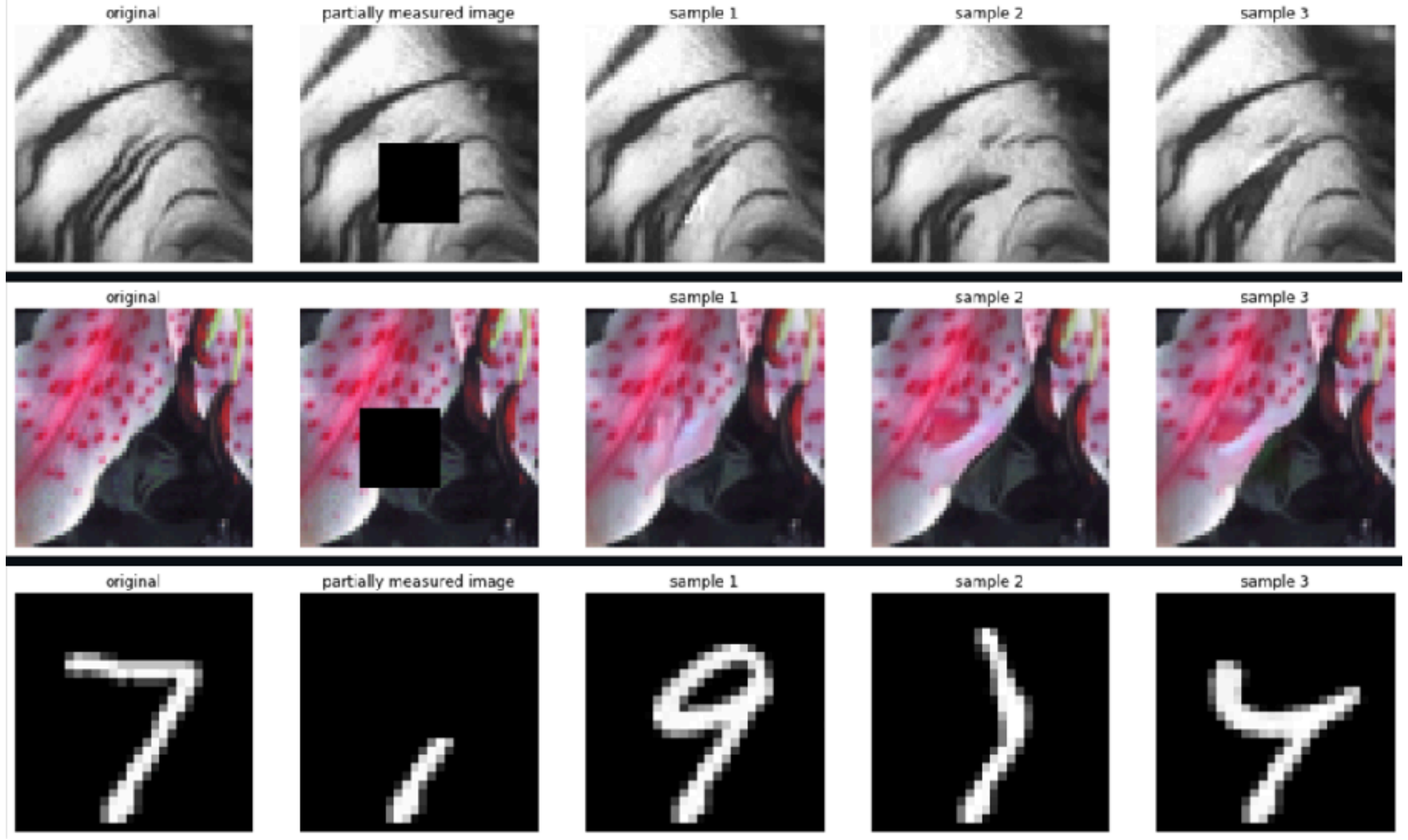
The second term is the gradient of the observation noise distribution, projected into the measurement space. If this is Gaussian with variance σ^2, it reduces to $M(y^c - x^c)$. The first term is the gradient of a function defined only within the subspace orthogonal to the measurements, and thus can be computed by projecting the measurement subspace out of the full gradient. Combining these gives:

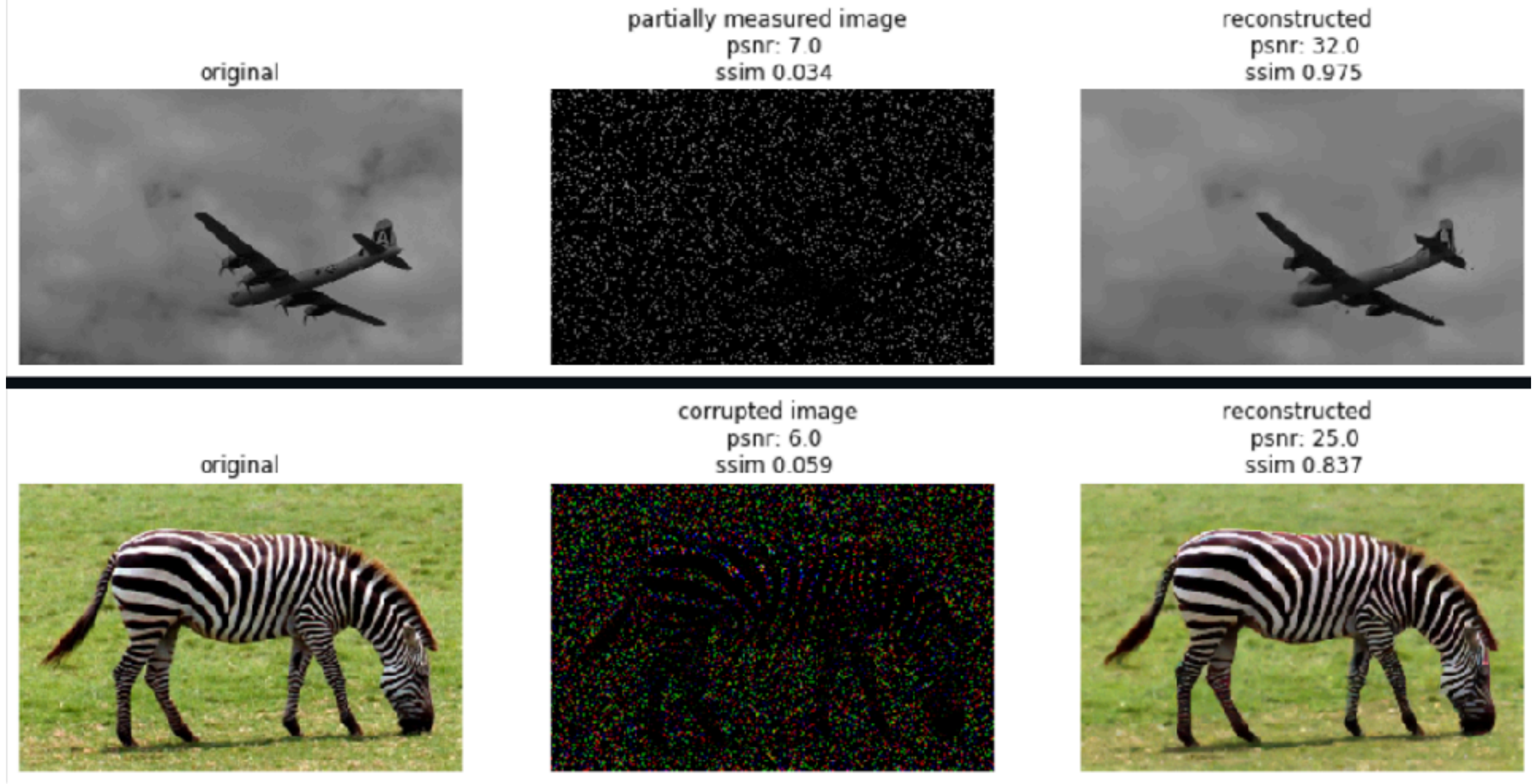
$$\begin{aligned} \sigma^2\nabla_y \log p(y) &= (I - MM^T)\sigma^2\nabla_y \log p(y) + M(x^c - y^c) \\ &= (I - MM^T)f(y) + M(x^c - M^T y). \end{aligned} \quad (9)$$

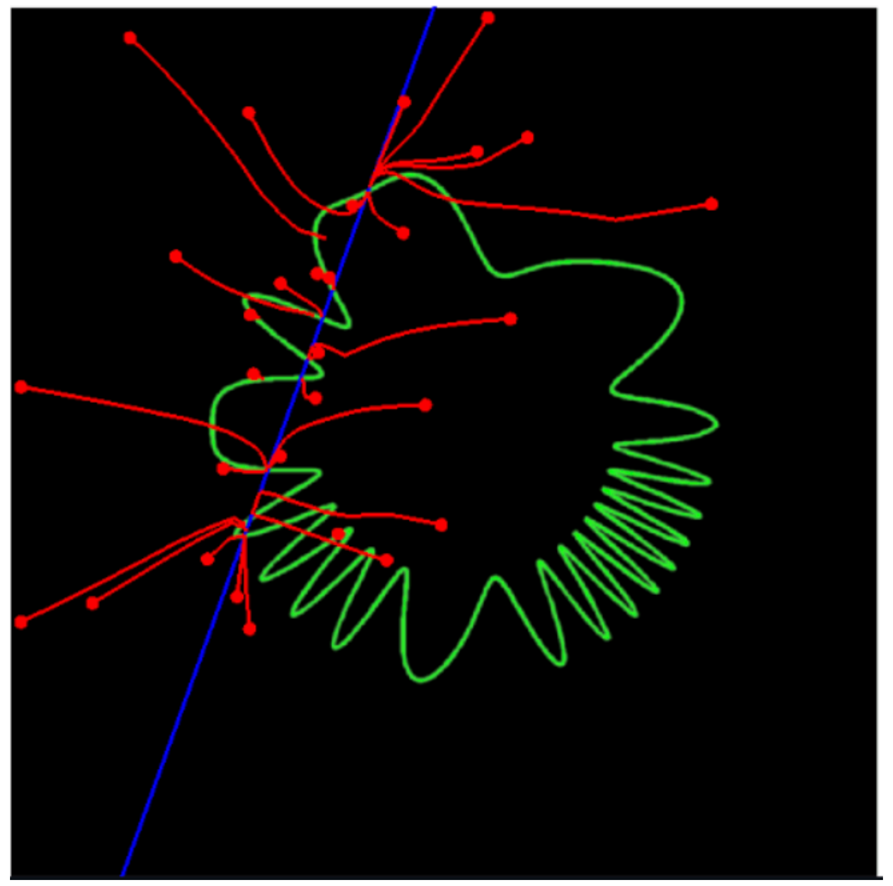


Algorithm 2: Coarse-to-fine stochastic ascent method for sampling on the residual of a denoiser, $f(y) = \hat{x}(y) - y$. Note: e is an imag

```
parameters: $\sigma_0$, $\sigma_L$, $h_0$, $\beta$, $M$, $x^c$
initialization: t=1; draw $y_0 \sim \mathcal{N}(0.5(I - MM^T)e + Mx^c,\ \sigma_0^2 I)$
while $\sigma_{t-1} \leq \sigma_L$ do
    $h_t = \frac{h_0 t}{1+h_0(t-1)}$;
    $d_t = (I - MM^T)f(y_{t-1}) + M(x^c - M^T y_{t-1})$;
    $\sigma_t^2 = \frac{||d_t||^2}{N}$;
    $\gamma_t^2 = \left((1 - \beta h_t)^2 - (1 - h_t)^2\right)\sigma_t^2$;
    Draw $z_t \sim \mathcal{N}(0, I)$;
    $y_t \leftarrow y_{t-1} + h_t d_t + \gamma_t z_t$;
    $t \leftarrow t + 1$
end
```







Images from: https://github.com/LabForComputationalVision/universal_inverse_problem

https://www.carmin.tv/en/collections/a-multiscale-tour-of-harmonic-analysis-and-machine-learning-to-celebrate-stephane-mallats-60th-birth/video/photographic-image-priors-in-the-era-of-machine-learning

Kadkhodaie, Z., & Simoncelli, E. P. (2021). Solving Linear Inverse Problems Using the Prior Implicit in a Denoiser (arXiv:2007.13640). arXiv. http://arxiv.org/abs/2007.13640

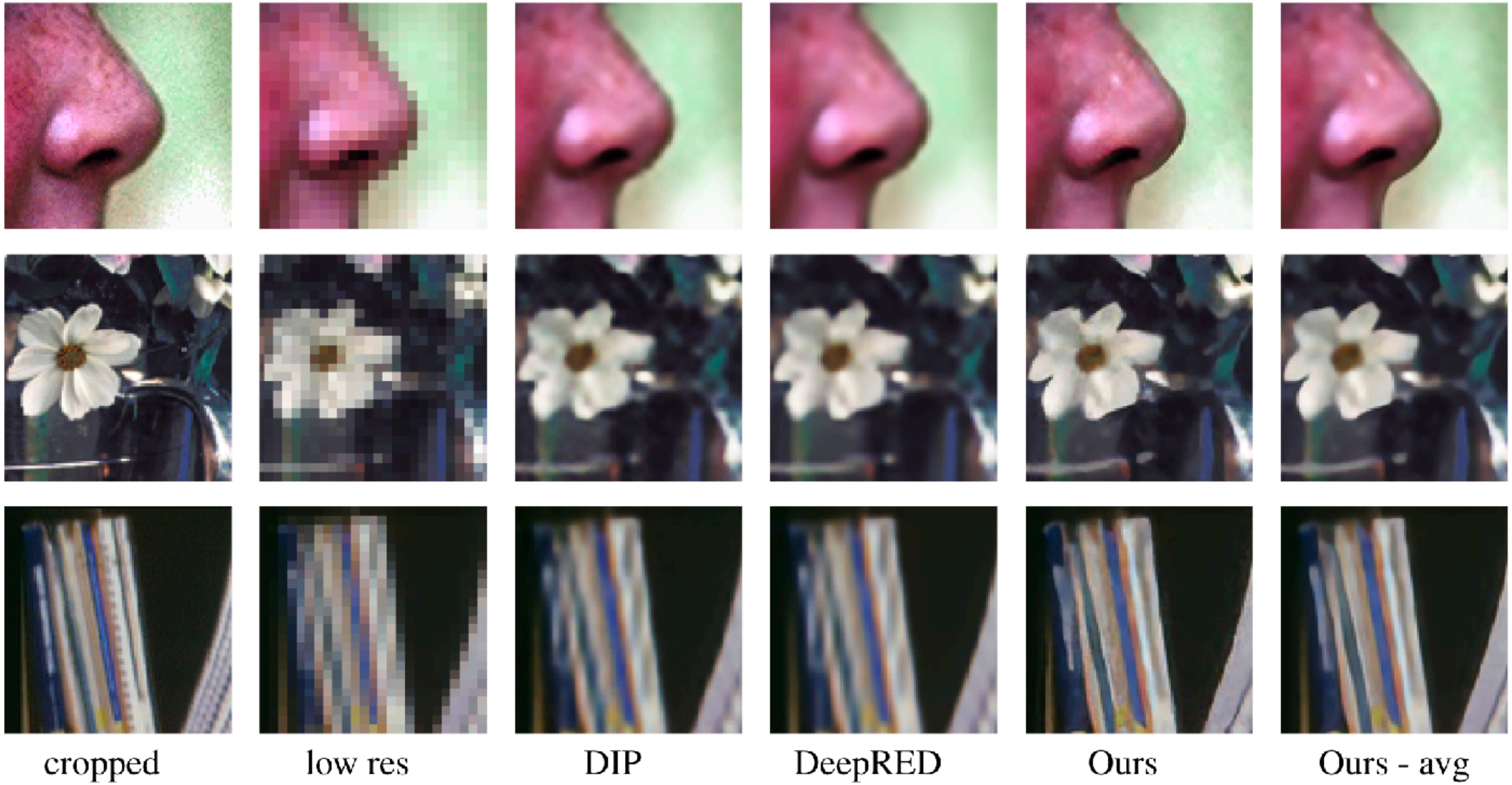


Figure 6: Spatial super-resolution. First column show three original images (cropped portion). Second column shows cropped portion with resolution reduced by averaging over 4x4 blocks (dimensionality reduction to 6.25%). Next three columns show reconstruction results obtained using DIP [8], DeepRED [14], and our method. Last column shows an average over 10 samples from our method, which is blurrier but with better PSNR (see Tables 1 and 2).

verse_problem

https://w ıge-priors-in-the-era-
of-machir

Kadkhodaie, Z., & Simoncelli, E. P. (2021). Solving Linear Inverse Problems Using the Prior Implicit in a Denoiser (arXiv:2007.13640). arXiv. http://arxiv.org/abs/2007.13640

Related ongoing work

very cool papers for future seminar sessions ;)

5/5

GENERALIZATION IN DIFFUSION MODELS ARISES FROM GEOMETRY-ADAPTIVE HARMONIC REPRESENTATION

Zahra Kadkhodaie
Ctr. for Data Science, New York University
zk388@nyu.edu

Florentin Guth
Ctr. for Data Science, New York University
Flatiron Institute, Simons Foundation
florentin.guth@nyu.edu

Eero P. Simoncelli
New York University
Flatiron Institute, Simons Foundation
eero.simoncelli@nyu.edu

Stéphane Mallat
Collège de France
Flatiron Institute, Simons Foundation
stephane.mallat@ens.fr

ABSTRACT

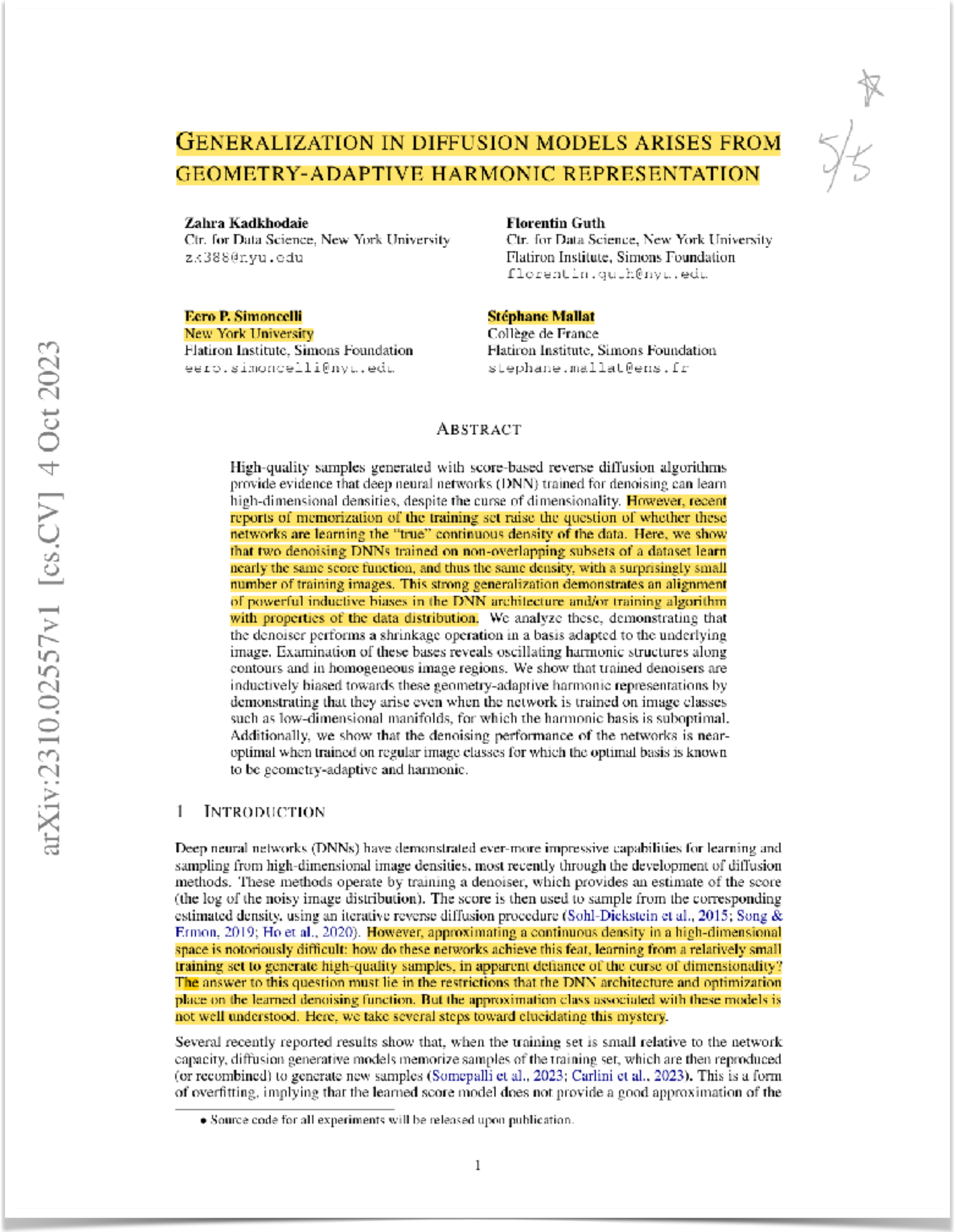
High-quality samples generated with score-based reverse diffusion algorithms provide evidence that deep neural networks (DNN) trained for denoising can learn high-dimensional densities, despite the curse of dimensionality. However, recent reports of memorization of the training set raise the question of whether these networks are learning the "true" continuous density of the data. Here, we show that two denoising DNNs trained on non-overlapping subsets of a dataset learn nearly the same score function, and thus the same density, with a surprisingly small number of training images. This strong generalization demonstrates an alignment of powerful inductive biases in the DNN architecture and/or training algorithm with properties of the data distribution. We analyze these, demonstrating that the denoiser performs a shrinkage operation in a basis adapted to the underlying image. Examination of these bases reveals oscillating harmonic structures along contours and in homogeneous image regions. We show that trained denoisers are inductively biased towards these geometry-adaptive harmonic representations by demonstrating that they arise even when the network is trained on image classes such as low-dimensional manifolds, for which the harmonic basis is suboptimal. Additionally, we show that the denoising performance of the networks is near-optimal when trained on regular image classes for which the optimal basis is known to be geometry-adaptive and harmonic.

1 INTRODUCTION

Deep neural networks (DNNs) have demonstrated ever-more impressive capabilities for learning and sampling from high-dimensional image densities, most recently through the development of diffusion methods. These methods operate by training a denoiser, which provides an estimate of the score (the log of the noisy image distribution). The score is then used to sample from the corresponding estimated density, using an iterative reverse diffusion procedure (Sohl-Dickstein et al., 2015; Song & Ermon, 2019; Ho et al., 2020). However, approximating a continuous density in a high-dimensional space is notoriously difficult: how do these networks achieve this feat, learning from a relatively small training set to generate high-quality samples, in apparent defiance of the curse of dimensionality? The answer to this question must lie in the restrictions that the DNN architecture and optimization place on the learned denoising function. But the approximation class associated with these models is not well understood. Here, we take several steps toward elucidating this mystery.

Several recently reported results show that, when the training set is small relative to the network capacity, diffusion generative models memorize samples of the training set, which are then reproduced (or recombined) to generate new samples (Somepalli et al., 2023; Carlini et al., 2023). This is a form of overfitting, implying that the learned score model does not provide a good approximation of the

• Source code for all experiments will be released upon publication.



arXiv:2310.02557v1 [cs.CV] 4 Oct 2023

5/5

GENERALIZATION IN DIFFUSION MODELS ARISES FROM GEOMETRY-ADAPTIVE HARMONIC REPRESENTATION

Zahra Kadkhodaie
Ctr. for Data Science, New York University
zk388@nyu.edu

Florentin Guth
Ctr. for Data Science, New York University
Flatiron Institute, Simons Foundation
florentin.guth@nyu.edu

Eero P. Simoncelli
New York University
Flatiron Institute, Simons Foundation
eero.simoncelli@nyu.edu

Stéphane Mallat
Collège de France
Flatiron Institute, Simons Foundation
stephane.mallat@ens.fr

ABSTRACT

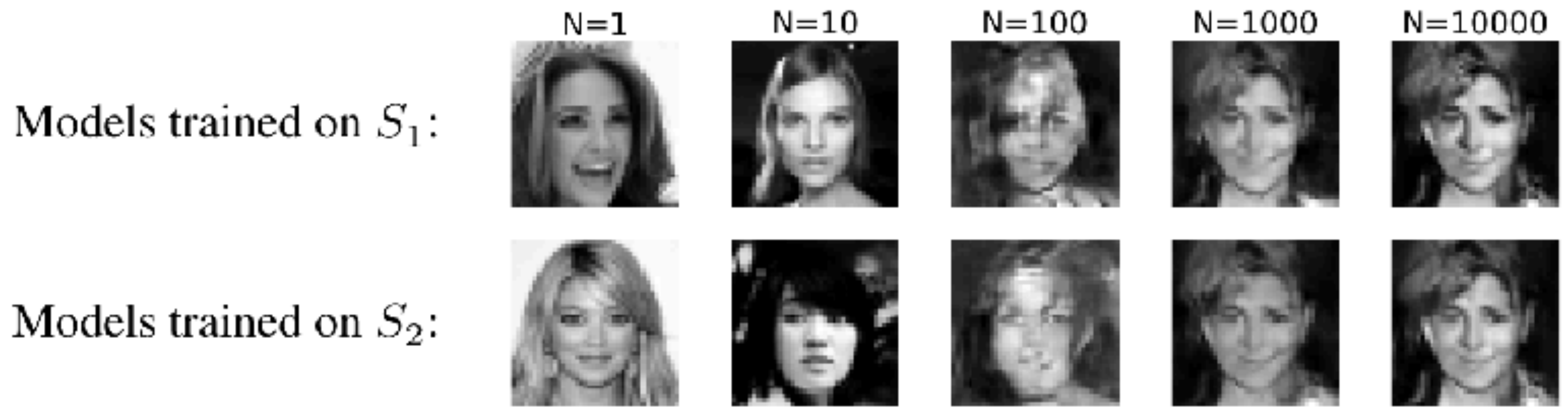
High-quality samples generated with score-based reverse diffusion algorithms provide evidence that deep neural networks (DNN) trained for denoising can learn high-dimensional densities, despite the curse of dimensionality. However, recent reports of memorization of the training set raise the question of whether these networks are learning the "true" continuous density of the data. Here, we show that two denoising DNNs trained on non-overlapping subsets of a dataset learn nearly the same score function, and thus the same density, with a surprisingly small number of training images. This strong generalization demonstrates an alignment of powerful inductive biases in the DNN architecture and/or training algorithm with properties of the data distribution. We analyze these, demonstrating that the denoiser performs a shrinkage operation in a basis adapted to the underlying image. Examination of these bases reveals oscillating harmonic structures along contours and in homogeneous image regions. We show that trained denoisers are inductively biased towards these geometry-adaptive harmonic representations by demonstrating that they arise even when the network is trained on image classes such as low-dimensional manifolds, for which the harmonic basis is suboptimal. Additionally, we show that the denoising performance of the networks is near-optimal when trained on regular image classes for which the optimal basis is known to be geometry-adaptive and harmonic.

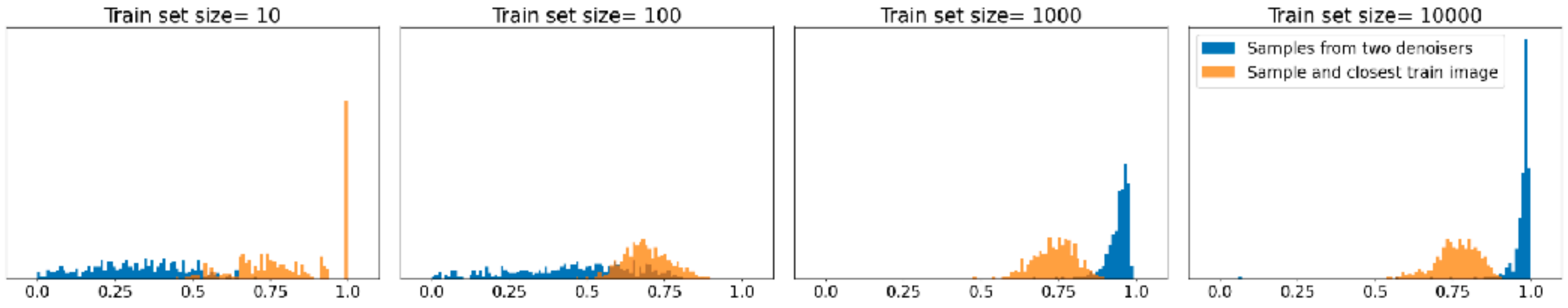
1 INTRODUCTION

Deep neural networks (DNNs) have demonstrated ever-more impressive capabilities for learning and sampling from high-dimensional image densities, most recently through the development of diffusion methods. These methods operate by training a denoiser, which provides an estimate of the score (the log of the noisy image distribution). The score is then used to sample from the corresponding estimated density, using an iterative reverse diffusion procedure (Sohl-Dickstein et al., 2015; Song & Ermon, 2019; Ho et al., 2020). However, approximating a continuous density in a high-dimensional space is notoriously difficult: how do these networks achieve this feat, learning from a relatively small training set to generate high-quality samples, in apparent defiance of the curse of dimensionality? The answer to this question must lie in the restrictions that the DNN architecture and optimization place on the learned denoising function. But the approximation class associated with these models is not well understood. Here, we take several steps toward elucidating this mystery.

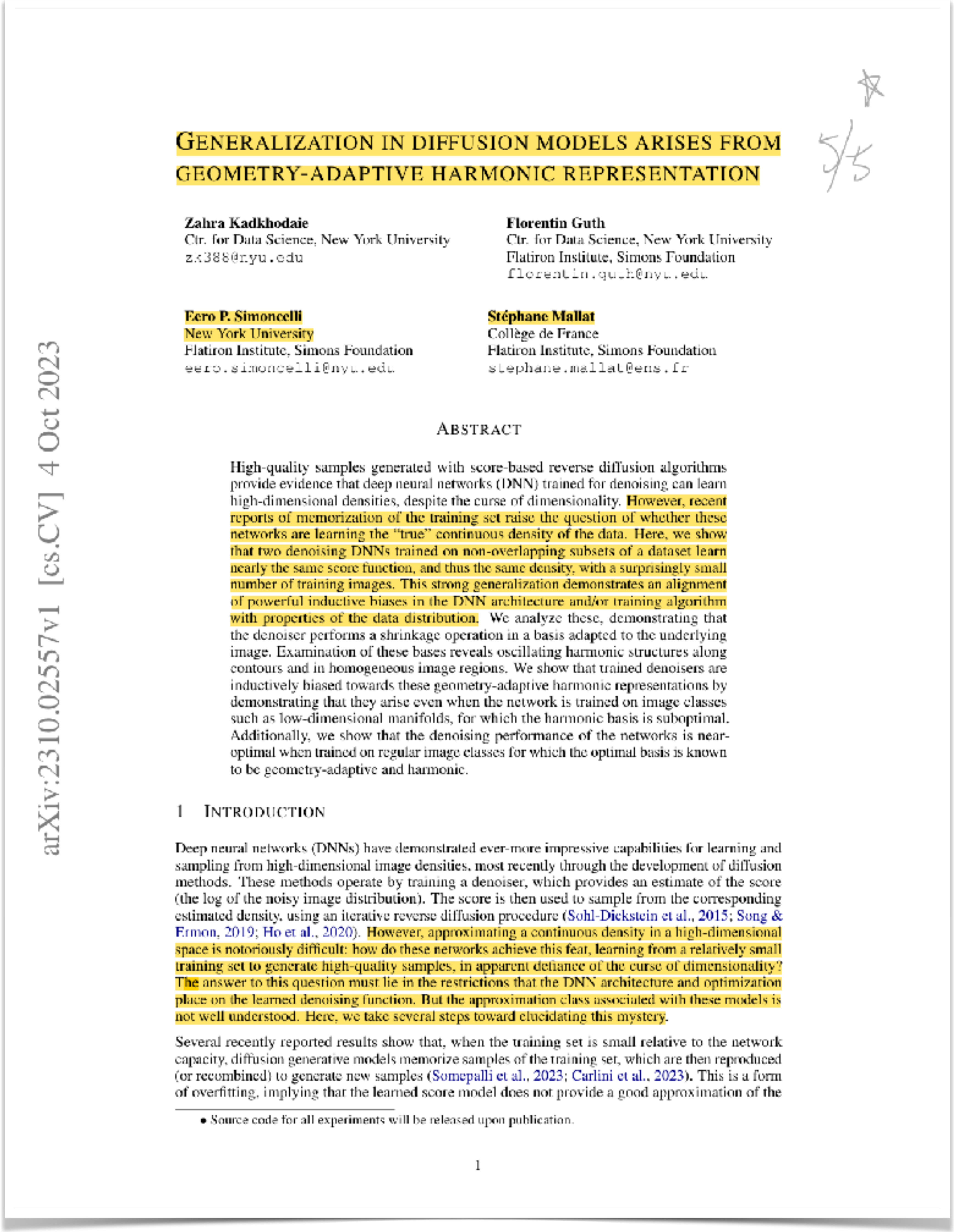
Several recently reported results show that, when the training set is small relative to the network capacity, diffusion generative models memorize samples of the training set, which are then reproduced (or recombined) to generate new samples (Somepalli et al., 2023; Carlini et al., 2023). This is a form of overfitting, implying that the learned score model does not provide a good approximation of the

• Source code for all experiments will be released upon publication.





Blue: Cosine similarity between samples generated from two (independently trained denoisers). Same initial point.
Orange: Cosine similarity between the generated sample and the closest samples in the training set.



5/5

Generalization in Diffusion Models Arises from Geometry-Adaptive Harmonic Representation

Zahra Kadkhodaie
Ctr. for Data Science, New York University
zk388@nyu.edu

Florentin Guth
Ctr. for Data Science, New York University
Flatiron Institute, Simons Foundation
florentin.guth@nyu.edu

Eero P. Simoncelli
New York University
Flatiron Institute, Simons Foundation
eero.simoncelli@nyu.edu

Stéphane Mallat
Collège de France
Flatiron Institute, Simons Foundation
stephane.mallat@ens.fr

Abstract

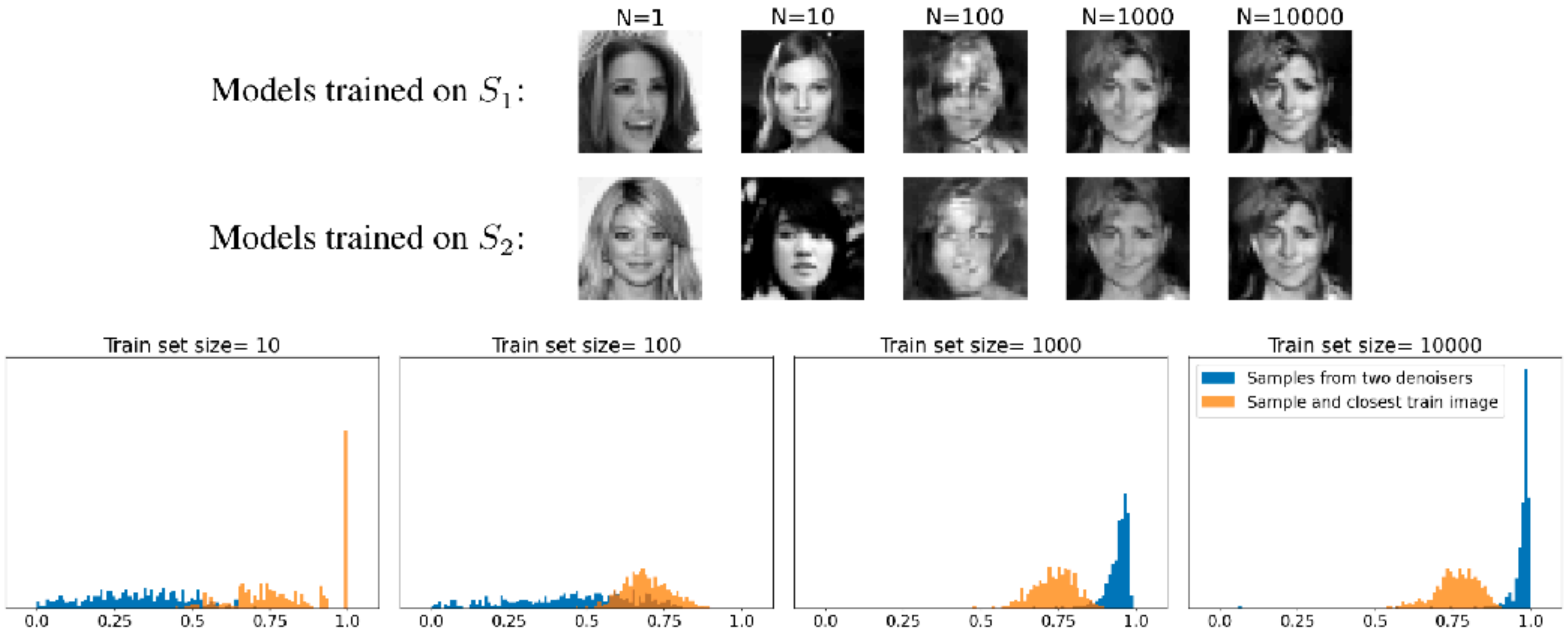
High-quality samples generated with score-based reverse diffusion algorithms provide evidence that deep neural networks (DNN) trained for denoising can learn high-dimensional densities, despite the curse of dimensionality. However, recent reports of memorization of the training set raise the question of whether these networks are learning the "true" continuous density of the data. Here, we show that two denoising DNNs trained on non-overlapping subsets of a dataset learn nearly the same score function, and thus the same density, with a surprisingly small number of training images. This strong generalization demonstrates an alignment of powerful inductive biases in the DNN architecture and/or training algorithm with properties of the data distribution. We analyze these, demonstrating that the denoiser performs a shrinkage operation in a basis adapted to the underlying image. Examination of these bases reveals oscillating harmonic structures along contours and in homogeneous image regions. We show that trained denoisers are inductively biased towards these geometry-adaptive harmonic representations by demonstrating that they arise even when the network is trained on image classes such as low-dimensional manifolds, for which the harmonic basis is suboptimal. Additionally, we show that the denoising performance of the networks is near-optimal when trained on regular image classes for which the optimal basis is known to be geometry-adaptive and harmonic.

1 Introduction

Deep neural networks (DNNs) have demonstrated ever-more impressive capabilities for learning and sampling from high-dimensional image densities, most recently through the development of diffusion methods. These methods operate by training a denoiser, which provides an estimate of the score (the log of the noisy image distribution). The score is then used to sample from the corresponding estimated density, using an iterative reverse diffusion procedure (Sohl-Dickstein et al., 2015; Song & Ermon, 2019; Ho et al., 2020). However, approximating a continuous density in a high-dimensional space is notoriously difficult: how do these networks achieve this feat, learning from a relatively small training set to generate high-quality samples, in apparent defiance of the curse of dimensionality? The answer to this question must lie in the restrictions that the DNN architecture and optimization place on the learned denoising function. But the approximation class associated with these models is not well understood. Here, we take several steps toward elucidating this mystery.

Several recently reported results show that, when the training set is small relative to the network capacity, diffusion generative models memorize samples of the training set, which are then reproduced (or recombined) to generate new samples (Somepalli et al., 2023; Carlini et al., 2023). This is a form of overfitting, implying that the learned score model does not provide a good approximation of the

- Source code for all experiments will be released upon publication.



Blue: Cosine similarity between samples generated from two (independently trained denoisers). Same initial point.
Orange: Cosine similarity between the generated sample and the closest samples in the training set.

3 Inductive Biases

The number of samples needed for estimation of an arbitrary probability density grows exponentially with dimensionality. As a result, estimating high-dimensional distributions without a strong implicit prior over the hypothesis space is infeasible. The network architecture and the optimization algorithm induce a prior over the space of functions. This implicit prior over the hypothesis space is refereed to as inductive bias (Wilson & Izmailov, 2020). In Section 2.2, we demonstrated that DNN denoisers can learn scores, and thus a density, from relatively small train sets. This generalization result is evidence

5/5

Generalization in Diffusion Models Arises from Geometry-Adaptive Harmonic Representation

Zahra Kadkhodaie
Ctr. for Data Science, New York University
zk388@nyu.edu

Florentin Guth
Ctr. for Data Science, New York University
Flatiron Institute, Simons Foundation
florentin.guth@nyu.edu

Eero P. Simoncelli
New York University
Flatiron Institute, Simons Foundation
eero.simoncelli@nyu.edu

Stéphane Mallat
Collège de France
Flatiron Institute, Simons Foundation
stephane.mallat@ens.fr

Abstract

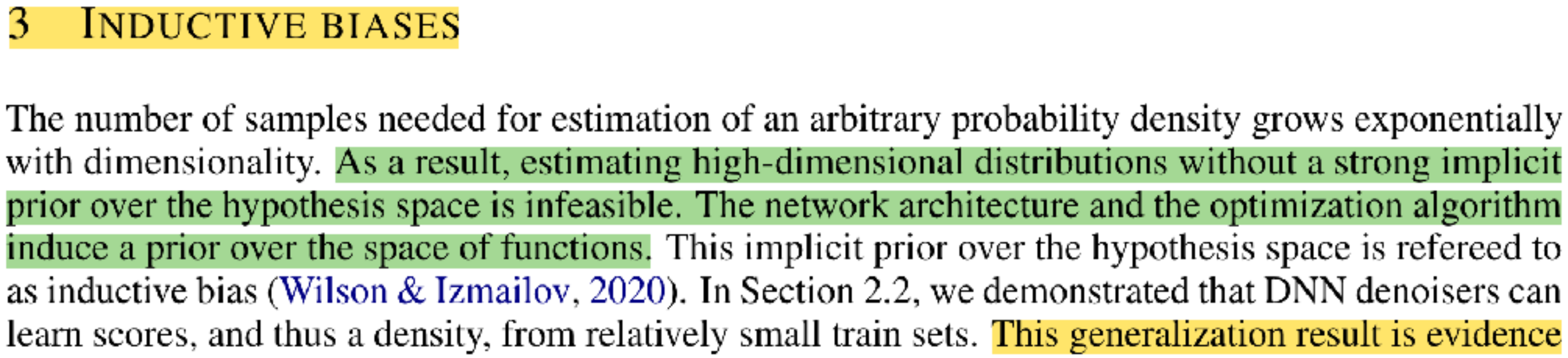
High-quality samples generated with score-based reverse diffusion algorithms provide evidence that deep neural networks (DNN) trained for denoising can learn high-dimensional densities, despite the curse of dimensionality. However, recent reports of memorization of the training set raise the question of whether these networks are learning the "true" continuous density of the data. Here, we show that two denoising DNNs trained on non-overlapping subsets of a dataset learn nearly the same score function, and thus the same density, with a surprisingly small number of training images. This strong generalization demonstrates an alignment of powerful inductive biases in the DNN architecture and/or training algorithm with properties of the data distribution. We analyze these, demonstrating that the denoiser performs a shrinkage operation in a basis adapted to the underlying image. Examination of these bases reveals oscillating harmonic structures along contours and in homogeneous image regions. We show that trained denoisers are inductively biased towards these geometry-adaptive harmonic representations by demonstrating that they arise even when the network is trained on image classes such as low-dimensional manifolds, for which the harmonic basis is suboptimal. Additionally, we show that the denoising performance of the networks is near-optimal when trained on regular image classes for which the optimal basis is known to be geometry-adaptive and harmonic.

1 Introduction

Deep neural networks (DNNs) have demonstrated ever-more impressive capabilities for learning and sampling from high-dimensional image densities, most recently through the development of diffusion methods. These methods operate by training a denoiser, which provides an estimate of the score (the log of the noisy image distribution). The score is then used to sample from the corresponding estimated density, using an iterative reverse diffusion procedure (Sohl-Dickstein et al., 2015; Song & Ermon, 2019; Ho et al., 2020). However, approximating a continuous density in a high-dimensional space is notoriously difficult: how do these networks achieve this feat, learning from a relatively small training set to generate high-quality samples, in apparent defiance of the curse of dimensionality? The answer to this question must lie in the restrictions that the DNN architecture and optimization place on the learned denoising function. But the approximation class associated with these models is not well understood. Here, we take several steps toward elucidating this mystery.

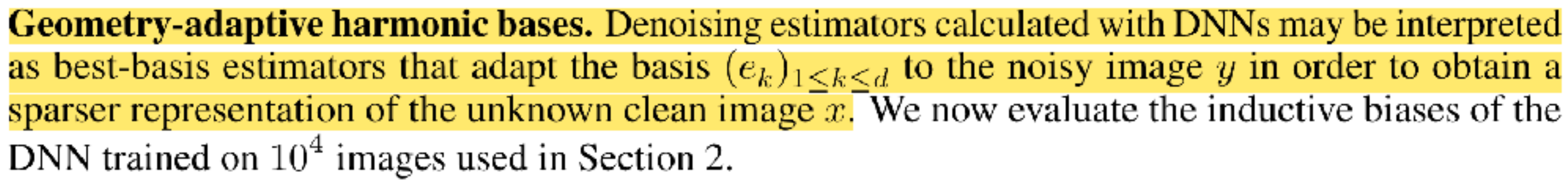
Several recently reported results show that, when the training set is small relative to the network capacity, diffusion generative models memorize samples of the training set, which are then reproduced (or recombined) to generate new samples (Somepalli et al., 2023; Carlini et al., 2023). This is a form of overfitting, implying that the learned score model does not provide a good approximation of the

- Source code for all experiments will be released upon publication.



3 Inductive Biases

The number of samples needed for estimation of an arbitrary probability density grows exponentially with dimensionality. As a result, estimating high-dimensional distributions without a strong implicit prior over the hypothesis space is infeasible. The network architecture and the optimization algorithm induce a prior over the space of functions. This implicit prior over the hypothesis space is refereed to as inductive bias (Wilson & Izmailov, 2020). In Section 2.2, we demonstrated that DNN denoisers can learn scores, and thus a density, from relatively small train sets. This generalization result is evidence



Geometry-adaptive harmonic bases. Denoising estimators calculated with DNNs may be interpreted as best-basis estimators that adapt the basis $(e_k)_{1\leq k\leq d}$ to the noisy image y in order to obtain a sparser representation of the unknown clean image x. We now evaluate the inductive biases of the DNN trained on 10^4 images used in Section 2.

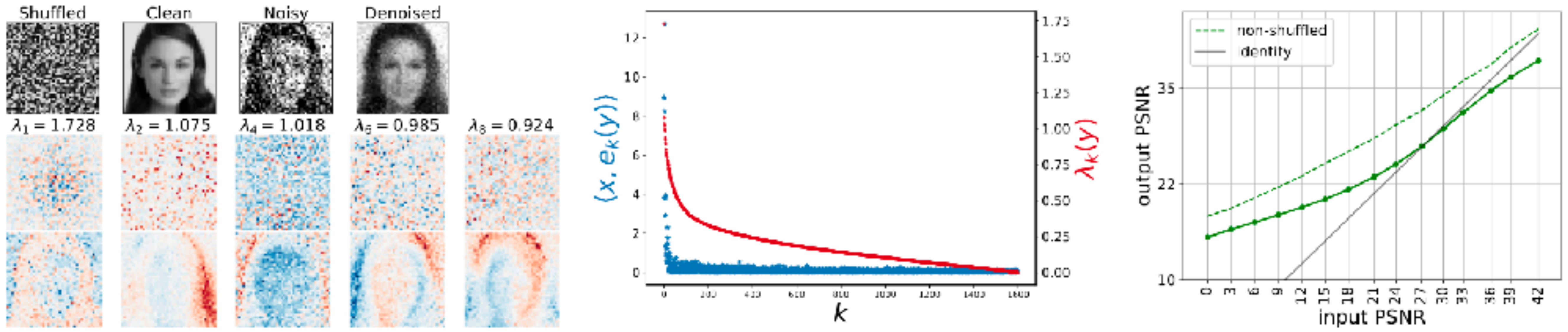
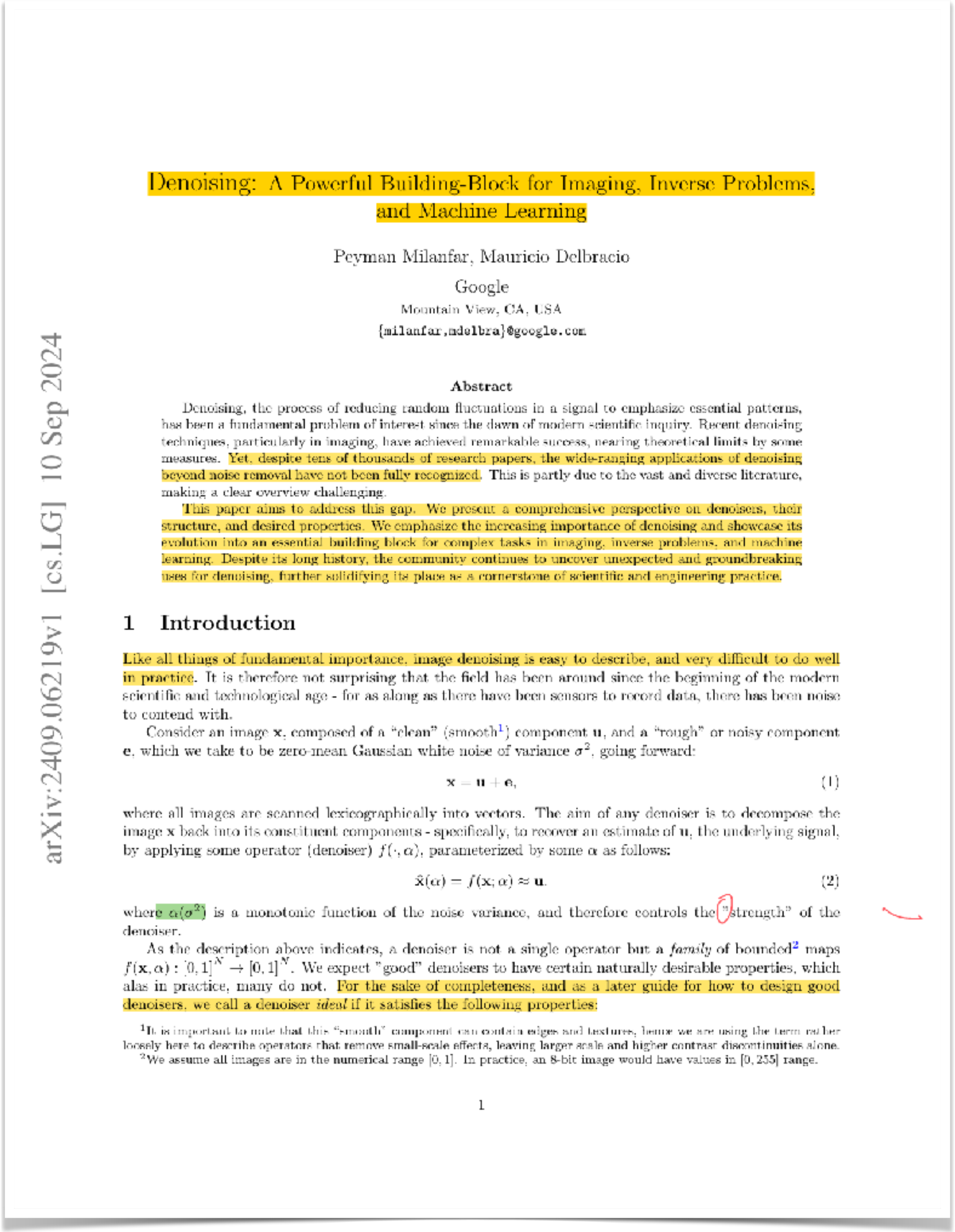


Figure 6: DNN denoiser trained on a shuffled face dataset. For visualization purposes, we "unshuffle" the pixels by applying the inverse of the permutation to the images before display. **Top left.** Clean (shuffled then unshuffled), noisy (unshuffled, $\sigma = 0.3$), and denoised (unshuffled) images. **Middle.** The shrinkage factors $\lambda_k(y)$ decay more slowly than when the denoiser is trained on non-shuffled faces (Figure 3). **Right.** The denoiser performs significantly worse than the denoiser trained on unshuffled faces. **Bottom left.** Basis vectors (top row: shuffled, bottom row: unshuffled). After unshuffling, we observe GAHBs adapted to the geometry of the face, although these are noisier and less precisely aligned with the image features than the non-shuffled examples in Figure 3.



Denoising: A Powerful Building-Block for Imaging, Inverse Problems, and Machine Learning

Peyman Milanfar, Mauricio Delbracio
Google
Mountain View, CA, USA
{milanfar,mdelbra}@google.com

Abstract

Denoising, the process of reducing random fluctuations in a signal to emphasize essential patterns, has been a fundamental problem of interest since the dawn of modern scientific inquiry. Recent denoising techniques, particularly in imaging, have achieved remarkable success, nearing theoretical limits by some measures. Yet, despite tens of thousands of research papers, the wide-ranging applications of denoising beyond noise removal have not been fully recognized. This is partly due to the vast and diverse literature, making a clear overview challenging.

This paper aims to address this gap. We present a comprehensive perspective on denoisers, their structure, and desired properties. We emphasize the increasing importance of denoising and showcase its evolution into an essential building block for complex tasks in imaging, inverse problems, and machine learning. Despite its long history, the community continues to uncover unexpected and groundbreaking uses for denoising, further solidifying its place as a cornerstone of scientific and engineering practice.

1 Introduction

Like all things of fundamental importance, image denoising is easy to describe, and very difficult to do well in practice. It is therefore not surprising that the field has been around since the beginning of the modern scientific and technological age - for as along as there have been sensors to record data, there has been noise to contend with.

Consider an image $\mathbf{x}$, composed of a "clean" (smooth[1]) component $\mathbf{u}$, and a "rough" or noisy component $\mathbf{e}$, which we take to be zero-mean Gaussian white noise of variance σ^2, going forward:

$$\mathbf{x} = \mathbf{u} + \mathbf{e}, \tag{1}$$

where all images are scanned lexicographically into vectors. The aim of any denoiser is to decompose the image $\mathbf{x}$ back into its constituent components - specifically, to recover an estimate of $\mathbf{u}$, the underlying signal, by applying some operator (denoiser) $f(\cdot, \alpha)$, parameterized by some α as follows:

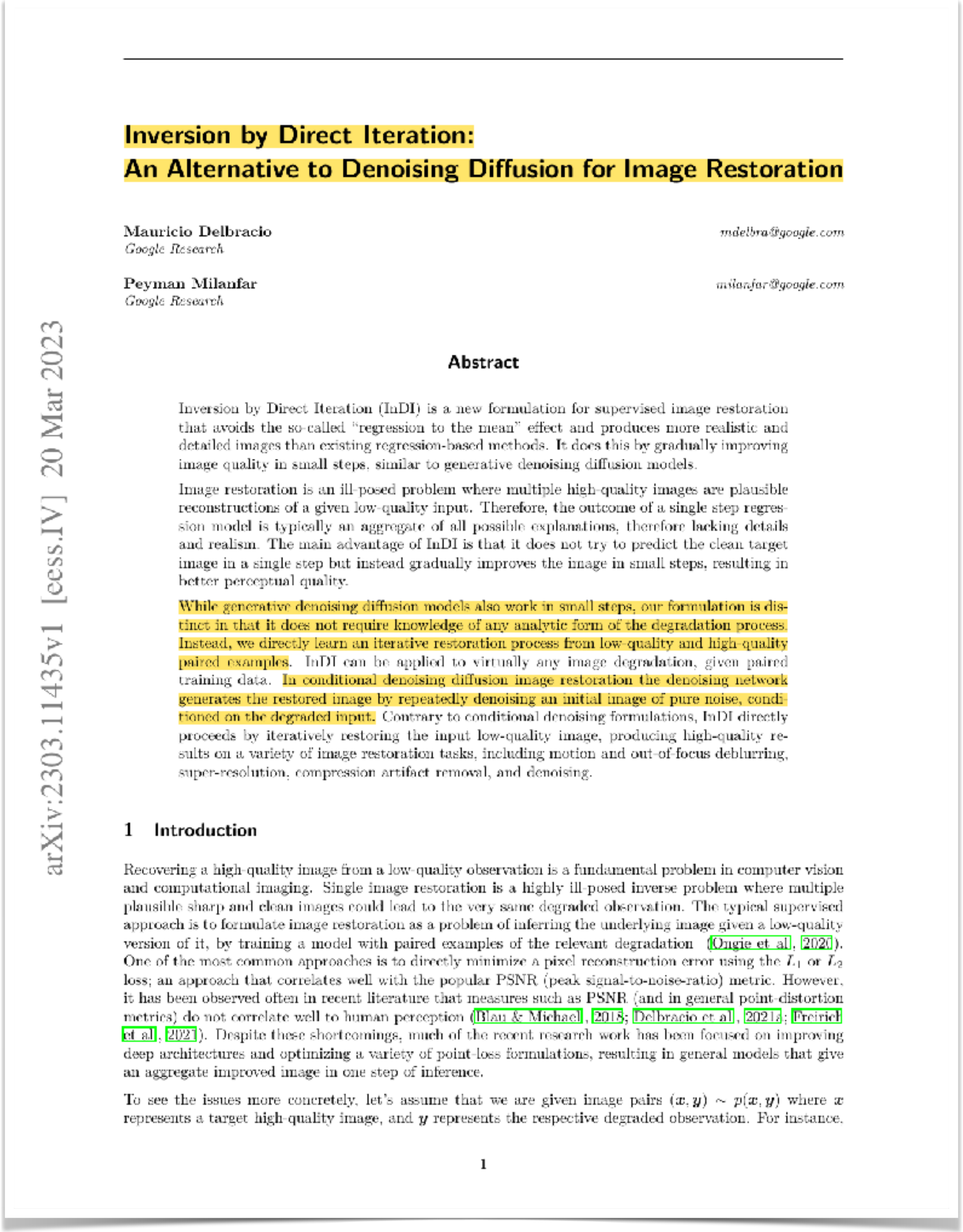
$$\hat{\mathbf{x}}(\alpha) = f(\mathbf{x}; \alpha) \approx \mathbf{u}. \tag{2}$$

where $\alpha(\sigma^2)$ is a monotonic function of the noise variance, and therefore controls the "strength" of the denoiser.

As the description above indicates, a denoiser is not a single operator but a *family* of bounded[2] maps $f(\mathbf{x}, \alpha) : [0,1]^N \to [0,1]^N$. We expect "good" denoisers to have certain naturally desirable properties, which alas in practice, many do not. For the sake of completeness, and as a later guide for how to design good denoisers, we call a denoiser *ideal* if it satisfies the following properties:

[1] It is important to note that this "smooth" component can contain edges and textures, hence we are using the term rather loosely here to describe operators that remove small-scale effects, leaving larger scale and higher contrast discontinuities alone.

[2] We assume all images are in the numerical range [0, 1]. In practice, an 8-bit image would have values in [0, 255] range.



Inversion by Direct Iteration: An Alternative to Denoising Diffusion for Image Restoration

Mauricio Delbracio mdelbra@google.com
Google Research

Peyman Milanfar milanfar@google.com
Google Research

Abstract

Inversion by Direct Iteration (InDI) is a new formulation for supervised image restoration that avoids the so-called "regression to the mean" effect and produces more realistic and detailed images than existing regression-based methods. It does this by gradually improving image quality in small steps, similar to generative denoising diffusion models.

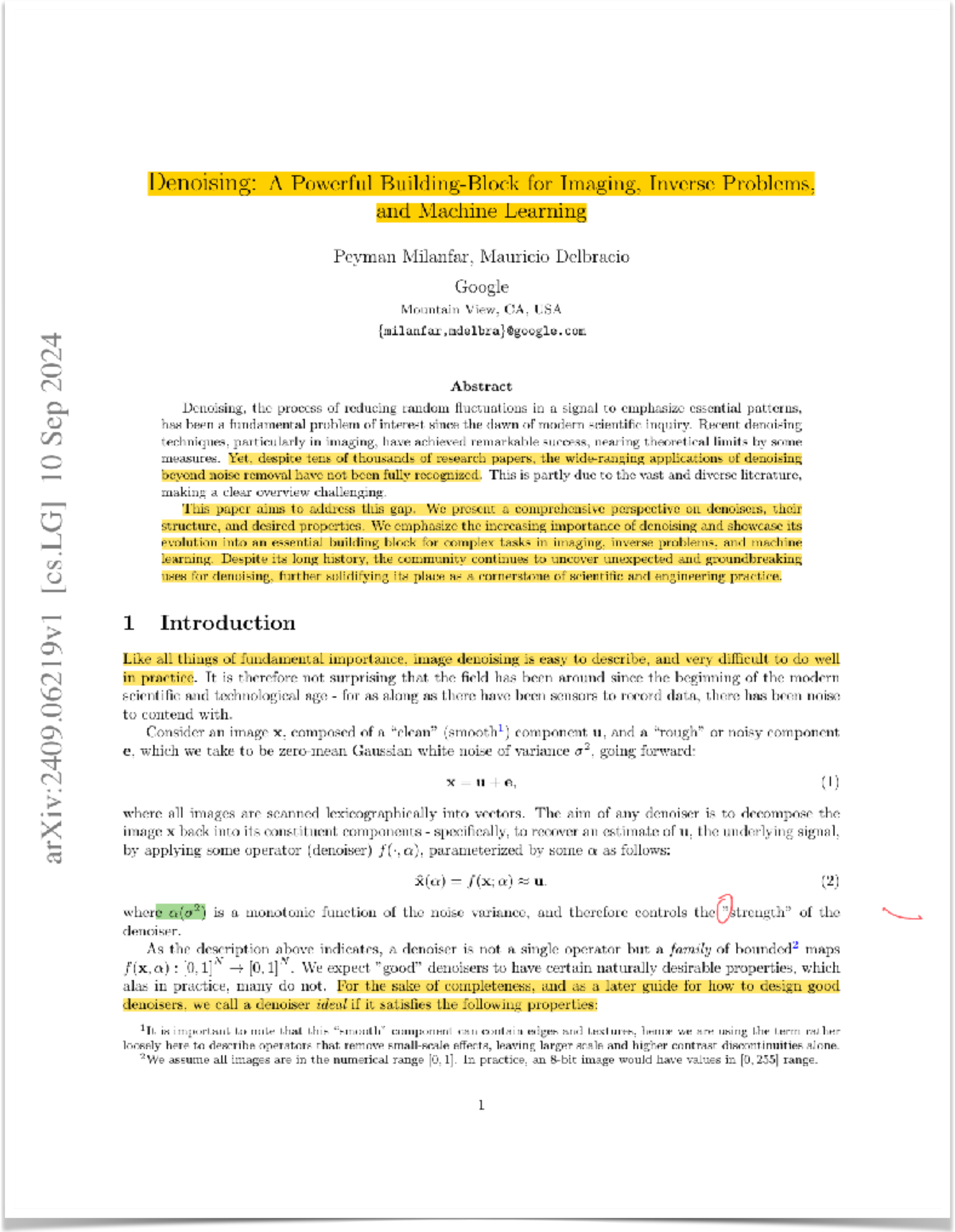
Image restoration is an ill-posed problem where multiple high-quality images are plausible reconstructions of a given low-quality input. Therefore, the outcome of a single step regression model is typically an aggregate of all possible explanations, therefore lacking details and realism. The main advantage of InDI is that it does not try to predict the clean target image in a single step but instead gradually improves the image in small steps, resulting in better perceptual quality.

While generative denoising diffusion models also work in small steps, our formulation is distinct in that it does not require knowledge of any analytic form of the degradation process. Instead, we directly learn an iterative restoration process from low-quality and high-quality paired examples. InDI can be applied to virtually any image degradation, given paired training data. In conditional denoising diffusion image restoration the denoising network generates the restored image by repeatedly denoising an initial image of pure noise, conditioned on the degraded input. Contrary to conditional denoising formulations, InDI directly proceeds by iteratively restoring the input low-quality image, producing high-quality results on a variety of image restoration tasks, including motion and out-of-focus deblurring, super-resolution, compression artifact removal, and denoising.

1 Introduction

Recovering a high-quality image from a low-quality observation is a fundamental problem in computer vision and computational imaging. Single image restoration is a highly ill-posed inverse problem where multiple plausible sharp and clean images could lead to the very same degraded observation. The typical supervised approach is to formulate image restoration as a problem of inferring the underlying image given a low-quality version of it, by training a model with paired examples of the relevant degradation (Ongie et al., 2020). One of the most common approaches is to directly minimize a pixel reconstruction error using the L_1 or L_2 loss; an approach that correlates well with the popular PSNR (peak signal-to-noise-ratio) metric. However, it has been observed often in recent literature that measures such as PSNR (and in general point-distortion metrics) do not correlate well to human perception (Blau & Michaeli, 2018; Delbracio et al., 2021a; Freirich et al., 2021). Despite these shortcomings, much of the recent research work has been focused on improving deep architectures and optimizing a variety of point-loss formulations, resulting in general models that give an aggregate improved image in one step of inference.

To see the issues more concretely, let's assume that we are given image pairs $(x, y) \sim p(x, y)$ where x represents a target high-quality image, and y represents the respective degraded observation. For instance,



Denoising: A Powerful Building-Block for Imaging, Inverse Problems, and Machine Learning

Peyman Milanfar, Mauricio Delbracio
Google
Mountain View, CA, USA
{milanfar,mdelbra}@google.com

Abstract

Denoising, the process of reducing random fluctuations in a signal to emphasize essential patterns, has been a fundamental problem of interest since the dawn of modern scientific inquiry. Recent denoising techniques, particularly in imaging, have achieved remarkable success, nearing theoretical limits by some measures. Yet, despite tens of thousands of research papers, the wide-ranging applications of denoising beyond noise removal have not been fully recognized. This is partly due to the vast and diverse literature, making a clear overview challenging.

This paper aims to address this gap. We present a comprehensive perspective on denoisers, their structure, and desired properties. We emphasize the increasing importance of denoising and showcase its evolution into an essential building block for complex tasks in imaging, inverse problems, and machine learning. Despite its long history, the community continues to uncover unexpected and groundbreaking uses for denoising, further solidifying its place as a cornerstone of scientific and engineering practice.

1 Introduction

Like all things of fundamental importance, image denoising is easy to describe, and very difficult to do well in practice. It is therefore not surprising that the field has been around since the beginning of the modern scientific and technological age - for as along as there have been sensors to record data, there has been noise to contend with.

Consider an image $\mathbf{x}$, composed of a "clean" (smooth[1]) component $\mathbf{u}$, and a "rough" or noisy component $\mathbf{e}$, which we take to be zero-mean Gaussian white noise of variance σ^2, going forward:

$$\mathbf{x} = \mathbf{u} + \mathbf{e}, \tag{1}$$

where all images are scanned lexicographically into vectors. The aim of any denoiser is to decompose the image $\mathbf{x}$ back into its constituent components - specifically, to recover an estimate of $\mathbf{u}$, the underlying signal, by applying some operator (denoiser) $f(\cdot, \alpha)$, parameterized by some α as follows:

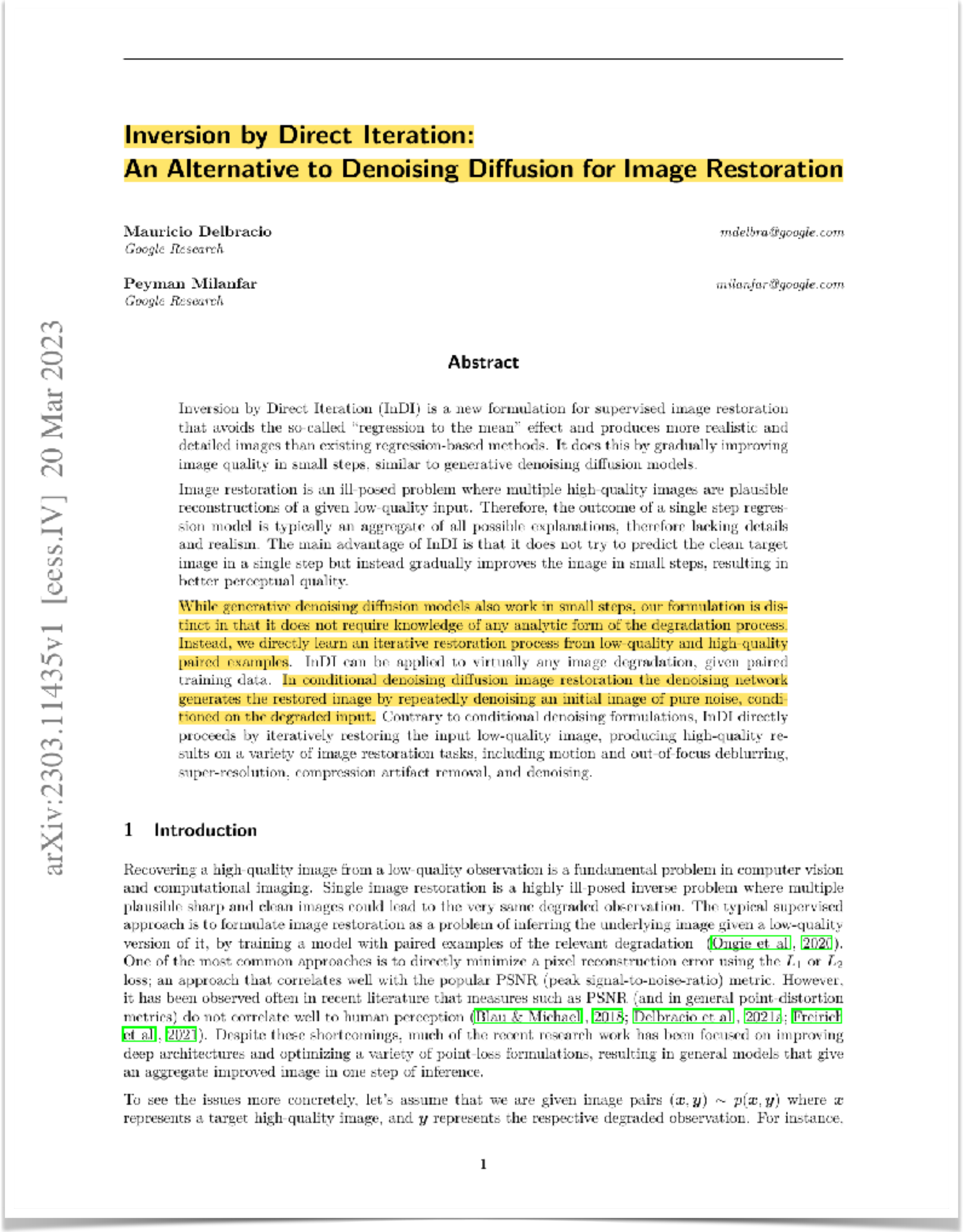
$$\hat{\mathbf{x}}(\alpha) = f(\mathbf{x}; \alpha) \approx \mathbf{u}. \tag{2}$$

where $\alpha(\sigma^2)$ is a monotonic function of the noise variance, and therefore controls the "strength" of the denoiser.

As the description above indicates, a denoiser is not a single operator but a *family* of bounded[2] maps $f(\mathbf{x}, \alpha) : [0,1]^N \rightarrow [0,1]^N$. We expect "good" denoisers to have certain naturally desirable properties, which alas in practice, many do not. For the sake of completeness, and as a later guide for how to design good denoisers, we call a denoiser *ideal* if it satisfies the following properties:

[1] It is important to note that this "smooth" component can contain edges and textures, hence we are using the term rather loosely here to describe operators that remove small-scale effects, leaving larger scale and higher contrast discontinuities alone.

[2] We assume all images are in the numerical range $[0, 1]$. In practice, an 8-bit image would have values in $[0, 255]$ range.



Inversion by Direct Iteration: An Alternative to Denoising Diffusion for Image Restoration

Mauricio Delbracio *mdelbra@google.com*
Google Research

Peyman Milanfar *milanfar@google.com*
Google Research

Abstract

Inversion by Direct Iteration (InDI) is a new formulation for supervised image restoration that avoids the so-called "regression to the mean" effect and produces more realistic and detailed images than existing regression-based methods. It does this by gradually improving image quality in small steps, similar to generative denoising diffusion models.

Image restoration is an ill-posed problem where multiple high-quality images are plausible reconstructions of a given low-quality input. Therefore, the outcome of a single step regression model is typically an aggregate of all possible explanations, therefore lacking details and realism. The main advantage of InDI is that it does not try to predict the clean target image in a single step but instead gradually improves the image in small steps, resulting in better perceptual quality.

While generative denoising diffusion models also work in small steps, our formulation is distinct in that it does not require knowledge of any analytic form of the degradation process. Instead, we directly learn an iterative restoration process from low-quality and high-quality paired examples. InDI can be applied to virtually any image degradation, given paired training data. In conditional denoising diffusion image restoration the denoising network generates the restored image by repeatedly denoising an initial image of pure noise, conditioned on the degraded input. Contrary to conditional denoising formulations, InDI directly proceeds by iteratively restoring the input low-quality image, producing high-quality results on a variety of image restoration tasks, including motion and out-of-focus deblurring, super-resolution, compression artifact removal, and denoising.

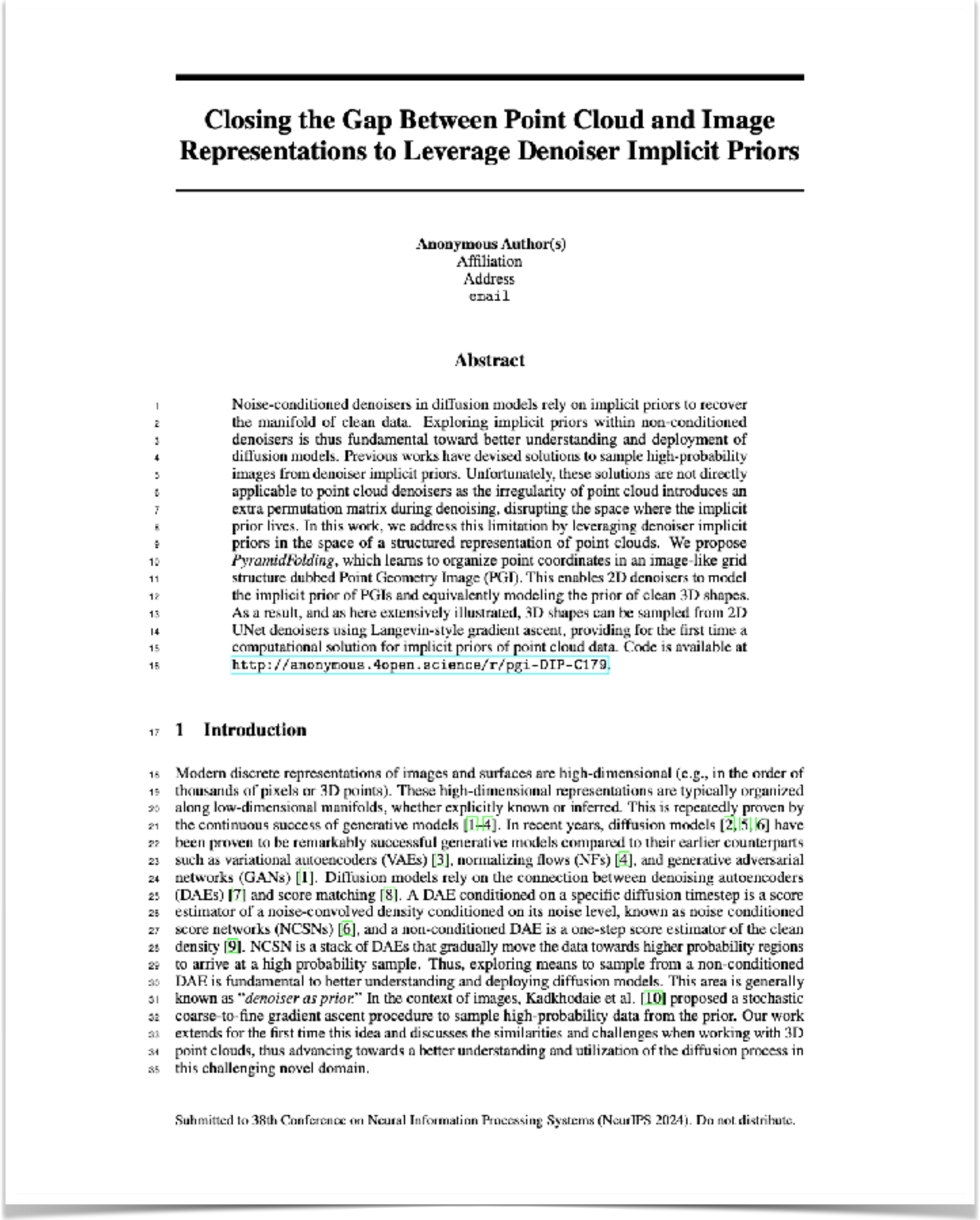
1 Introduction

Recovering a high-quality image from a low-quality observation is a fundamental problem in computer vision and computational imaging. Single image restoration is a highly ill-posed inverse problem where multiple plausible sharp and clean images could lead to the very same degraded observation. The typical supervised approach is to formulate image restoration as a problem of inferring the underlying image given a low-quality version of it, by training a model with paired examples of the relevant degradation (Ongie et al., 2020). One of the most common approaches is to directly minimize a pixel reconstruction error using the L_1 or L_2 loss; an approach that correlates well with the popular PSNR (peak signal-to-noise-ratio) metric. However, it has been observed often in recent literature that measures such as PSNR (and in general point-distortion metrics) do not correlate well to human perception (Blau & Michael, 2018; Delbracio et al., 2021a; Freirich et al., 2021). Despite these shortcomings, much of the recent research work has been focused on improving deep architectures and optimizing a variety of point-loss formulations, resulting in general models that give an aggregate improved image in one step of inference.

To see the issues more concretely, let's assume that we are given image pairs $(x, y) \sim p(x, y)$ where x represents a target high-quality image, and y represents the respective degraded observation. For instance,

Solving inverse problems using a denoiser

- “Plug-n-Play” - Venkatakrishnan, Bouman & Wohlberg (2013)
  Zhang et al. (2017);
  Romano, Elad, & Milanfar (2017) - “RED”;
  Kamilov, Mansour & Wohlberg (2019);
  Mataev Elad & Milanfar (2019);
  Reehorst & Schnitzer (2019);
  Sun, Liu & Kamilov (2019);
  Teodoro, Bioucas-Dias & Figueiredo (2019)
- “Denoising Score-matching” - Vincent (2011);
  Alain & Bengio (2014);
  Saremi & Hyvarinen (2019);
  Li, Chen, Sommers (2019);
  Guo, Han & Wen (2019);
  Song & Ermon (2019);
  Bigdeli et. al. (2020)
- “Diffusion-based generative models” - Sohl-Dickstein et al. (2015)



Closing the Gap Between Point Cloud and Image Representations to Leverage Denoiser Implicit Priors

Anonymous Author(s)
Affiliation
Address
email

Abstract

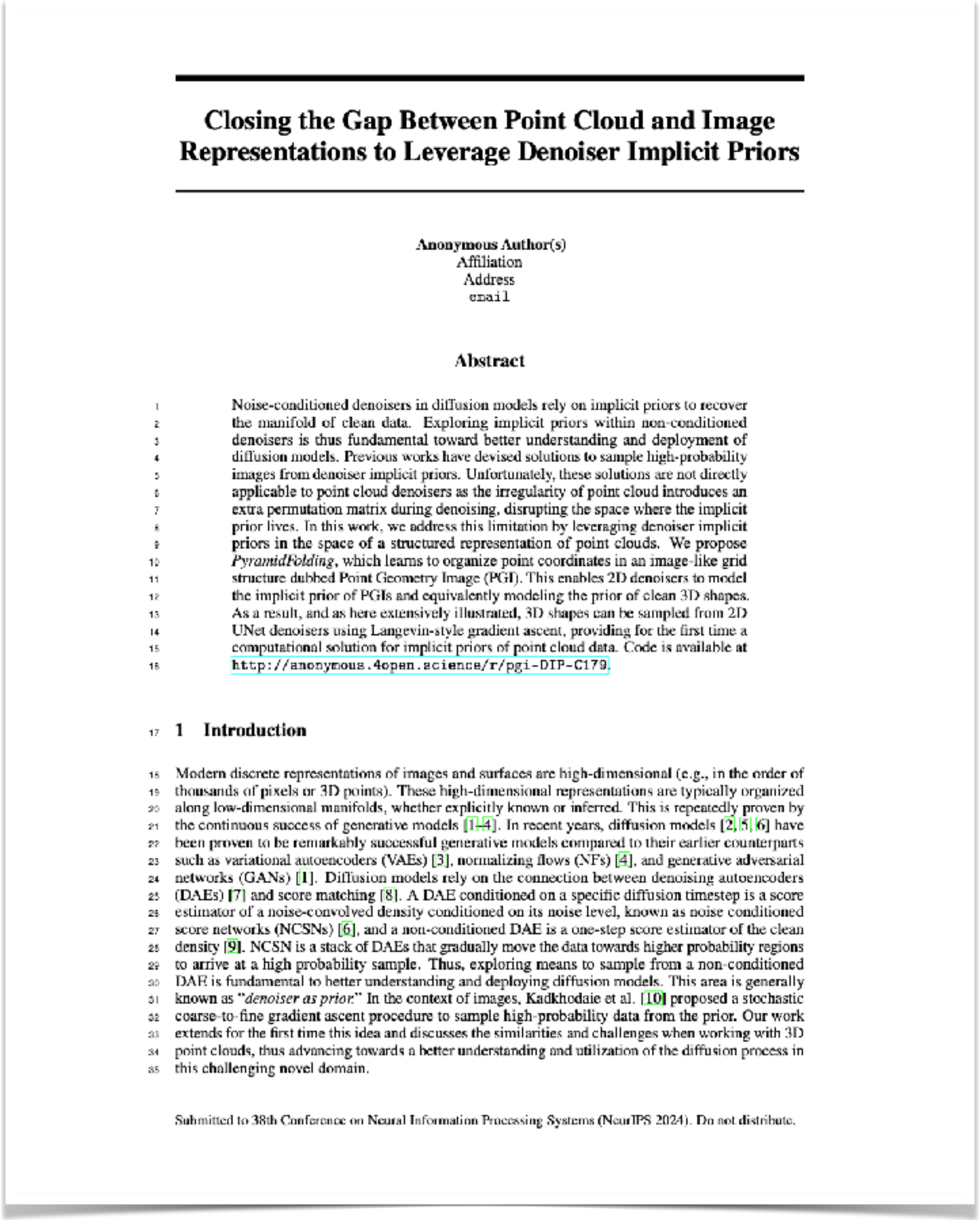
Noise-conditioned denoisers in diffusion models rely on implicit priors to recover the manifold of clean data. Exploring implicit priors within non-conditioned denoisers is thus fundamental toward better understanding and deployment of diffusion models. Previous works have devised solutions to sample high-probability images from denoiser implicit priors. Unfortunately, these solutions are not directly applicable to point cloud denoisers as the irregularity of point cloud introduces an extra permutation matrix during denoising, disrupting the space where the implicit prior lives. In this work, we address this limitation by leveraging denoiser implicit priors in the space of a structured representation of point clouds. We propose *PyramidFolding*, which learns to organize point coordinates in an image-like grid structure dubbed Point Geometry Image (PGI). This enables 2D denoisers to model the implicit prior of PGIs and equivalently modeling the prior of clean 3D shapes. As a result, and as here extensively illustrated, 3D shapes can be sampled from 2D UNet denoisers using Langevin-style gradient ascent, providing for the first time a computational solution for implicit priors of point cloud data. Code is available at `http://anonymous.4open.science/r/pgi-DIP-C179`.

1 Introduction

Modern discrete representations of images and surfaces are high-dimensional (e.g., in the order of thousands of pixels or 3D points). These high-dimensional representations are typically organized along low-dimensional manifolds, whether explicitly known or inferred. This is repeatedly proven by the continuous success of generative models [1–4]. In recent years, diffusion models [2, 5, 6] have been proven to be remarkably successful generative models compared to their earlier counterparts such as variational autoencoders (VAEs) [3], normalizing flows (NFs) [4], and generative adversarial networks (GANs) [1]. Diffusion models rely on the connection between denoising autoencoders (DAEs) [7] and score matching [8]. A DAE conditioned on a specific diffusion timestep is a score estimator of a noise-convolved density conditioned on its noise level, known as noise conditioned score networks (NCSNs) [6], and a non-conditioned DAE is a one-step score estimator of the clean density [9]. NCSN is a stack of DAEs that gradually move the data towards higher probability regions to arrive at a high probability sample. Thus, exploring means to sample from a non-conditioned DAE is fundamental to better understanding and deploying diffusion models. This area is generally known as "*denoiser as prior.*" In the context of images, Kadkhodaie et al. [10] proposed a stochastic coarse-to-fine gradient ascent procedure to sample high-probability data from the prior. Our work extends for the first time this idea and discusses the similarities and challenges when working with 3D point clouds, thus advancing towards a better understanding and utilization of the diffusion process in this challenging novel domain.

Submitted to 38th Conference on Neural Information Processing Systems (NeurIPS 2024). Do not distribute.

Yuzhou, Di Martino, Sapiro. "Closing the Gap Between Point Cloud and Image Representations to Leverage Denoiser Implicit Priors."



Closing the Gap Between Point Cloud and Image Representations to Leverage Denoiser Implicit Priors

Anonymous Author(s)
Affiliation
Address
`email`

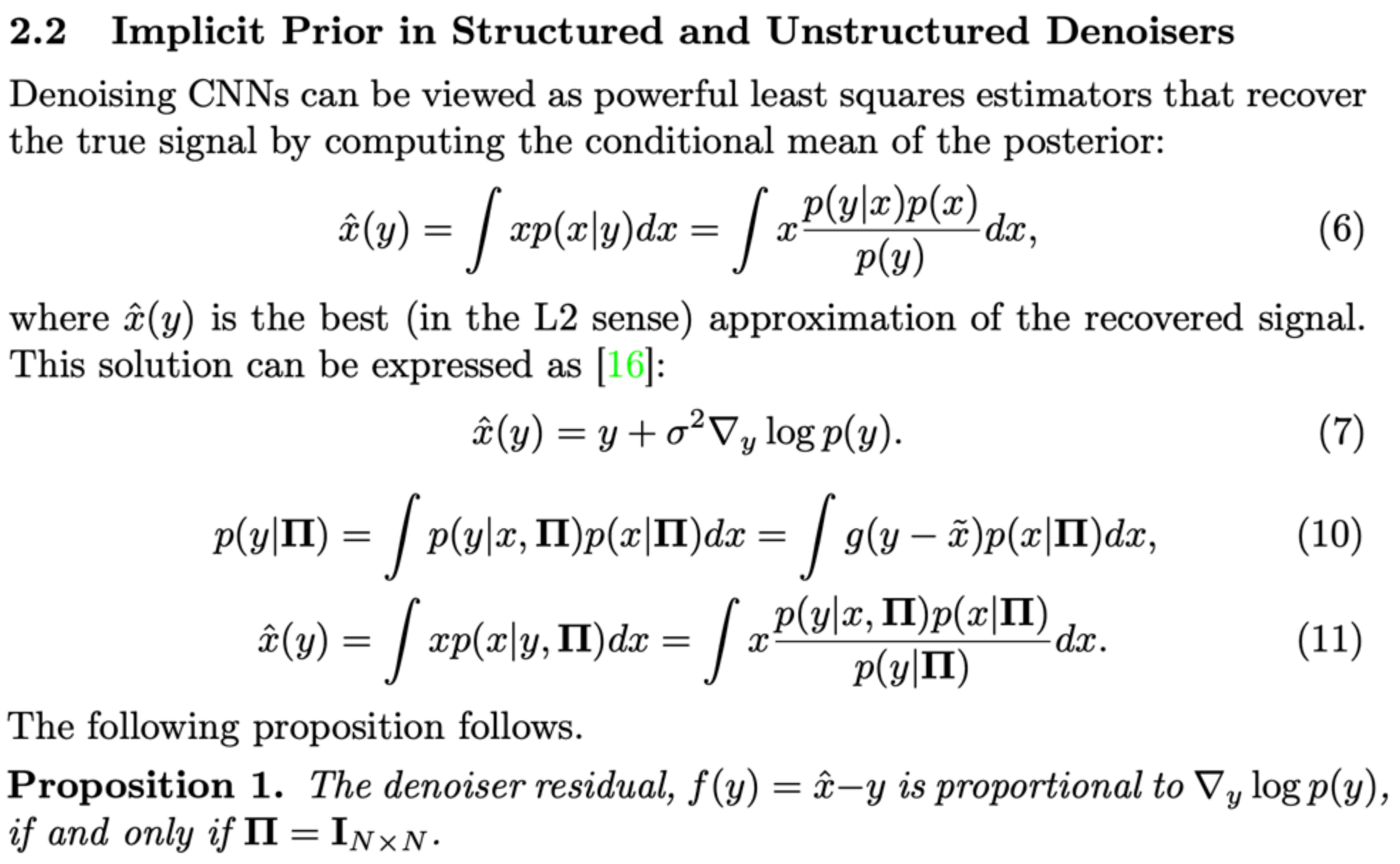
Abstract

Noise-conditioned denoisers in diffusion models rely on implicit priors to recover the manifold of clean data. Exploring implicit priors within non-conditioned denoisers is thus fundamental toward better understanding and deployment of diffusion models. Previous works have devised solutions to sample high-probability images from denoiser implicit priors. Unfortunately, these solutions are not directly applicable to point cloud denoisers as the irregularity of point cloud introduces an extra permutation matrix during denoising, disrupting the space where the implicit prior lives. In this work, we address this limitation by leveraging denoiser implicit priors in the space of a structured representation of point clouds. We propose *PyramidFolding*, which learns to organize point coordinates in an image-like grid structure dubbed Point Geometry Image (PGI). This enables 2D denoisers to model the implicit prior of PGIs and equivalently modeling the prior of clean 3D shapes. As a result, and as here extensively illustrated, 3D shapes can be sampled from 2D UNet denoisers using Langevin-style gradient ascent, providing for the first time a computational solution for implicit priors of point cloud data. Code is available at `http://anonymous.4open.science/r/pgi-DIP-C179`.

1 Introduction

Modern discrete representations of images and surfaces are high-dimensional (e.g., in the order of thousands of pixels or 3D points). These high-dimensional representations are typically organized along low-dimensional manifolds, whether explicitly known or inferred. This is repeatedly proven by the continuous success of generative models [1–4]. In recent years, diffusion models [2, 5, 6] have been proven to be remarkably successful generative models compared to their earlier counterparts such as variational autoencoders (VAEs) [3], normalizing flows (NFs) [4], and generative adversarial networks (GANs) [1]. Diffusion models rely on the connection between denoising autoencoders (DAEs) [7] and score matching [8]. A DAE conditioned on a specific diffusion timestep is a score estimator of a noise-convolved density conditioned on its noise level, known as noise conditioned score networks (NCSNs) [6], and a non-conditioned DAE is a one-step score estimator of the clean density [9]. NCSN is a stack of DAEs that gradually move the data towards higher probability regions to arrive at a high probability sample. Thus, exploring means to sample from a non-conditioned DAE is fundamental to better understanding and deploying diffusion models. This area is generally known as "*denoiser as prior.*" In the context of images, Kadkhodaie et al. [10] proposed a stochastic coarse-to-fine gradient ascent procedure to sample high-probability data from the prior. Our work extends for the first time this idea and discusses the similarities and challenges when working with 3D point clouds, thus advancing towards a better understanding and utilization of the diffusion process in this challenging novel domain.

Submitted to 38th Conference on Neural Information Processing Systems (NeurIPS 2024). Do not distribute.



2.2 Implicit Prior in Structured and Unstructured Denoisers

Denoising CNNs can be viewed as powerful least squares estimators that recover the true signal by computing the conditional mean of the posterior:

$$\hat{x}(y) = \int x p(x|y) dx = \int x \frac{p(y|x)p(x)}{p(y)} dx, \tag{6}$$

where $\hat{x}(y)$ is the best (in the L2 sense) approximation of the recovered signal. This solution can be expressed as [16]:

$$\hat{x}(y) = y + \sigma^2 \nabla_y \log p(y). \tag{7}$$

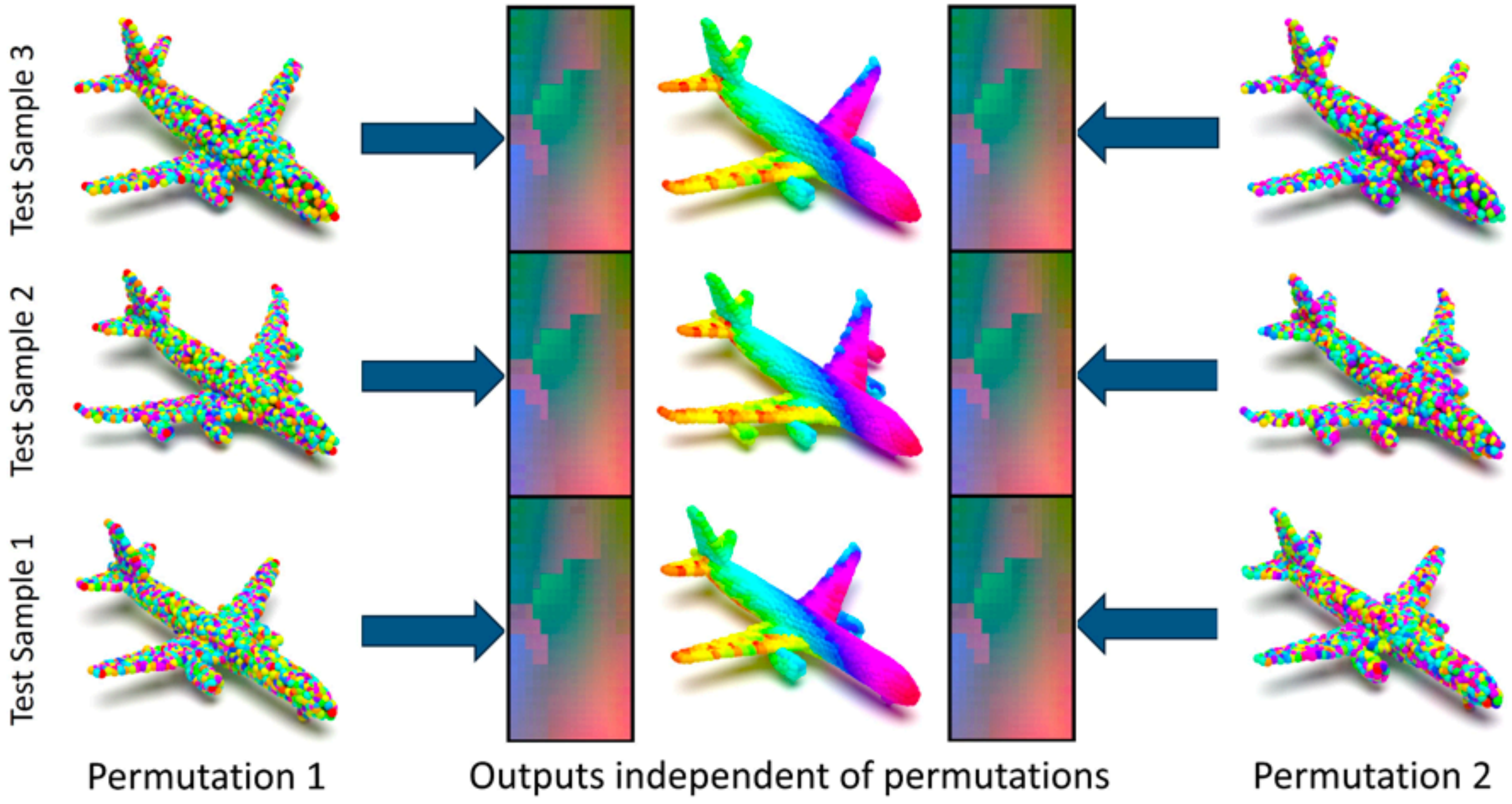
$$p(y|\mathbf{\Pi}) = \int p(y|x, \mathbf{\Pi}) p(x|\mathbf{\Pi}) dx = \int g(y - \tilde{x}) p(x|\mathbf{\Pi}) dx, \tag{10}$$

$$\hat{x}(y) = \int x p(x|y, \mathbf{\Pi}) dx = \int x \frac{p(y|x, \mathbf{\Pi}) p(x|\mathbf{\Pi})}{p(y|\mathbf{\Pi})} dx. \tag{11}$$

The following proposition follows.

Proposition 1. *The denoiser residual, $f(y) = \hat{x} - y$ is proportional to $\nabla_y \log p(y)$, if and only if $\mathbf{\Pi} = \mathbf{I}_{N \times N}$.*

Kadkhodaie, Zahra, and Eero Simoncelli. "Stochastic solutions for linear inverse problems using the prior implicit in a denoiser." *Advances in Neural Information Processing Systems* 34 (2021): 13242-13254.



2.2 Implicit Prior in Structured and Unstructured Denoisers

Denoising CNNs can be viewed as powerful least squares estimators that recover the true signal by computing the conditional mean of the posterior:

$$\hat{x}(y) = \int x p(x|y) dx = \int x \frac{p(y|x)p(x)}{p(y)} dx, \tag{6}$$

where $\hat{x}(y)$ is the best (in the L2 sense) approximation of the recovered signal. This solution can be expressed as [16]:

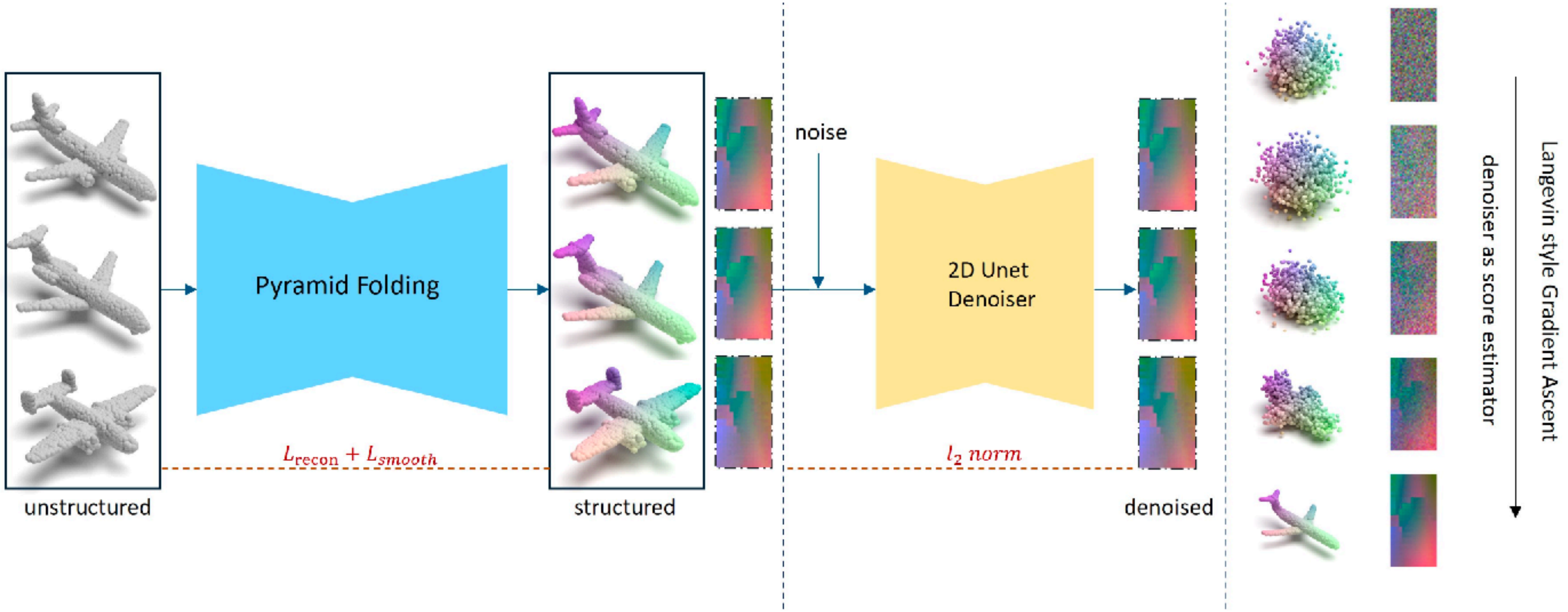
$$\hat{x}(y) = y + \sigma^2 \nabla_y \log p(y). \tag{7}$$

$$p(y|\mathbf{\Pi}) = \int p(y|x,\mathbf{\Pi}) p(x|\mathbf{\Pi}) dx = \int g(y - \tilde{x}) p(x|\mathbf{\Pi}) dx, \tag{10}$$

$$\hat{x}(y) = \int x p(x|y,\mathbf{\Pi}) dx = \int x \frac{p(y|x,\mathbf{\Pi}) p(x|\mathbf{\Pi})}{p(y|\mathbf{\Pi})} dx. \tag{11}$$

The following proposition follows.

Proposition 1. *The denoiser residual, $f(y) = \hat{x} - y$ is proportional to $\nabla_y \log p(y)$, if and only if $\mathbf{\Pi} = \mathbf{I}_{N \times N}$.*



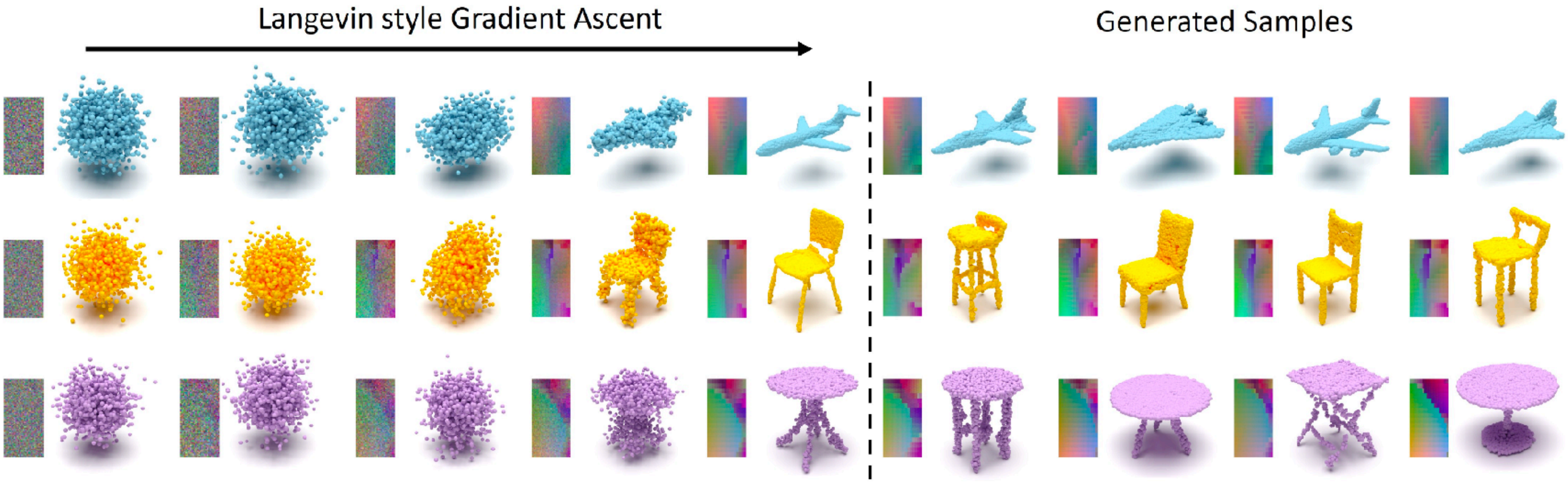


Fig. 5: Examples of single class generation. All samples are generated starting from standard Guassian noise and following Algorithm 1.

Milanfar, Peyman, and Mauricio Delbracio. "Denoising: A Powerful Building-Block for Imaging, Inverse Problems, and Machine Learning." *arXiv preprint arXiv:2409.06219* (2024).

Kadkhodaie, Zahra, and Eero Simoncelli. "Stochastic solutions for linear inverse problems using the prior implicit in a denoiser." *Advances in Neural Information Processing Systems* 34 (2021): 13242-13254.

Muchas gracias