# Chapter 2
# Variational Image Restoration

Variational Methods in Imaging

March 2023

Yvain QUÉAU
GREYC-CNRS
ENSICAEN - Université de Caen Normandie

# Overview

1. **Inverse Problems and Variational Methods**
2. **Variational Methods, Euler-Lagrange Equations and Diffusion**
3. **Numerical Solving of the Restoration Problem**

# Overview

# Inverse Problems, Ill-Posedness and Regularization

In mathematics, the conversion of measurement data into information about the observed object or the observed physical system is referred to as an inverse problem.

Following Hadamard (1902), a mathematical problem is called well-posed iff:

1. A solution exists.
2. The solution is unique.
3. The solution's behavior changes continuously with the initial conditions.

Inverse problems are often ill-posed. Since the measurement data is often not sufficient to uniquely characterize the observed object or system, one introduces prior knowledge to disambiguate which solutions are apriori more likely. In the context of variational methods this prior knowledge gives rise to the regularity term.

## Discrete VS Continuous Inverse Problems

Consider for instance a standard discrete inverse problem

$$y = Ax + \epsilon$$

with $y \in \mathbb{R}^m$ the observations, $x \in \mathbb{R}^n$ the signal to reconstruct, and $A \in \mathbb{R}^{m \times n}$ a known forward operator.

What can we say about the following cases in terms of well-posedness, and in terms of our capacity to reconstruct the "perfect" signal $x$ ?

- $m > n$, no noise $\epsilon$
- $m < n$, no noise $\epsilon$ (e.g., inpainting)
- $m = n$, Gaussian noise $\epsilon$ (e.g., denoising)

A standard way to solve such difficult inverse problems consists in resorting to regularization:

$$\min_x \quad E_{\text{data}}(y - Ax) + \lambda E_{\text{regul}}(x)$$

This framework can be extended to consider *functions* $y, x$ instead of vectors: this is the concept behind **variational methods**

# A Simple Example: Image Denoising

Let $f : \Omega \to \mathbb{R}$ be an input greylevel image corrupted by noise. The goal is to compute a denoised version $u$ of the image $f$.

The desired function $u$ should fulfill two criteria:

- The function $u$ should be similar to $f$.
- The function $u$ should be spatially smooth.

Both criteria can be combined in the following cost function (or energy):

$$E(u) = E_{data}(u, f) + \lambda\, E_{smoothness}(u),$$

where the first term measures the similarity of $u$ and $f$ and the second term measures the smoothness of $u$. A weighting or regularization parameter $\lambda \geq 0$ specifies the relative importance of smoothness versus data fit.

Most variational approaches have the above form. They merely differ in how the similarity term (data term) and the smoothness term (regularizer) are defined.

## Image Restoration: Denoising

Image restoration is a classical inverse problem: Given an observed image $f : \Omega \to \mathbb{R}$ and a (typically stochastic) model of an image degradation process, we want to restore the original image $u : \Omega \to \mathbb{R}$.

Image denoising is an example of image restoration where we assume that the true image $u$ is corrupted by (additive) noise:

$$f = u + \eta, \qquad \eta \sim \mathcal{N}(0, \sigma^2).$$

Rudin, Osher, Fatemi (1992) denoise $f$ by minimizing a quadratic data term with Total Variation (TV) regularization:

$$\min_{u\colon \Omega \to \mathbb{R}} \underbrace{\frac{1}{2} \iint_\Omega |u(x) - f(x)|^2 dx}_{E_{data}(u,f)} + \lambda \underbrace{\iint_\Omega \|\nabla u(x)\| dx}_{E_{smoothness}(u)}.$$

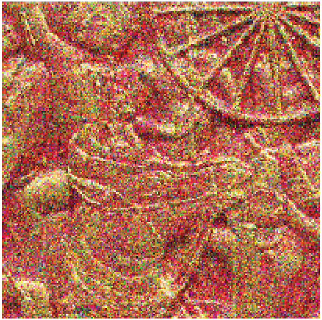
This gives rise to the necessary optimality condition (Euler-Lagrange equation):

$$u(x) - f(x) - \lambda \operatorname{div}\left( \frac{1}{\|\nabla u(x)\|} \nabla u(x) \right) = 0, \quad \forall x \in \Omega$$

Solving this PDE yields the denoised image.

# Image Restoration: Denoising

$$\hat{u} = \underset{u:\,\Omega \to \mathbb{R}}{\operatorname{argmin}} \frac{1}{2} \iint |u(x) - f(x)|^2 dx + \lambda \iint \|\nabla u(x)\| dx.$$

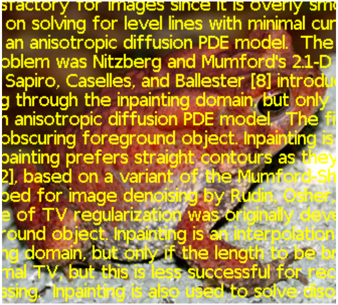


noisy ($f$)

denoised ($\hat{u}$)

# Image Restoration: Inpainting

Image inpainting is a particular image restoration technique which explicitly handles (interpolate and / or extrapolate) missing data.




Corrupted ($f$)

Denoised ($\hat{u}$)

Assume $f : \Omega \subset \mathbb{R}^2 \to \mathbb{R}$ a graylevel image, but only $\Omega_D \subset \Omega$ is "reliable". Then, denoising (or deblurring, etc.) should not use the $f$-data over $\Omega \backslash \Omega_D$. The standard TV-inpainting model is then:

$$\hat{u} = \underset{u:\,\Omega\to\mathbb{R}}{\operatorname{argmin}} \int_{\Omega_D} |u - f|^2 \, dx \; + \; \lambda \int_{\Omega} |\nabla u| \, dx.$$

# Limits of TV-inpainting

TV-inpainting is a very naive interpolation technique which does not transport texture...

## A Few Classic Fidelity Terms

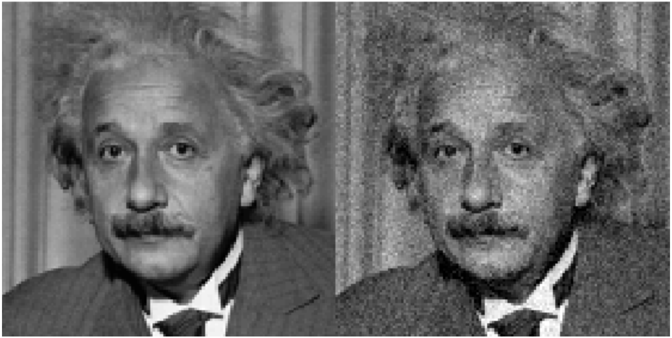
When modeling the restoration problem as the minimization of

$$E(u) = E_{data}(u, f) + \lambda\, E_{smoothness}(u),$$

several choices can be considered for the data term.

Gaussian noise is fine for modeling "small" perturbations. It yields the choice (smooth and convex):

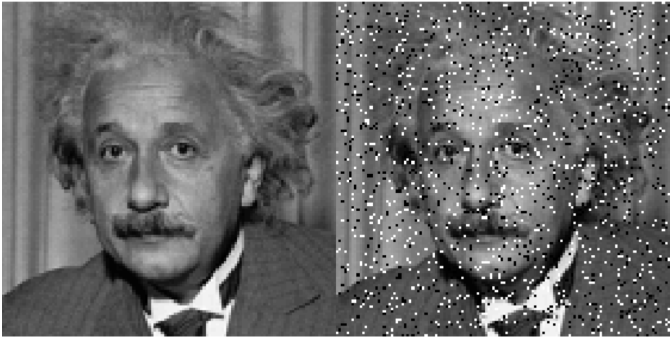
$$E_{data}(u, f) = \underbrace{\int |f(x) - u(x)|^2 \,\mathrm{d}x}_{:=\|f-u\|_2^2}$$



# A Few Classic Fidelity Terms

Laplace noise is better for modeling "impulsive" perturbations. It yields the choice (convex, yet non-smooth):

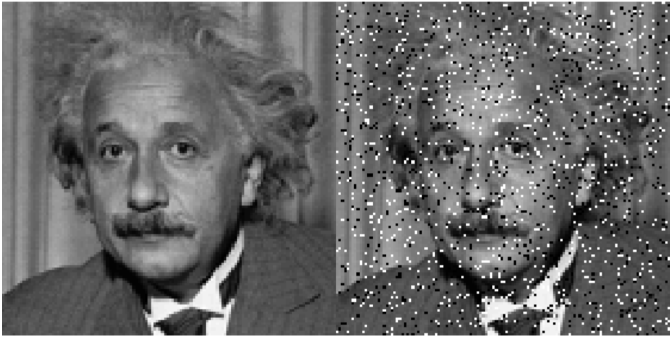
$$E_{data}(u, f) = \underbrace{\min_{u} \int |f(x) - u(x)| \, \mathrm{d}x}_{:=\|f-u\|_1}$$



# A Few Classic Fidelity Terms

Cauchy noise is even better for modeling “impulsive” perturbations. It yields the choice (smooth, yet non-convex):

$$E_{data}(u, f) = \int \log \left( |f(x) - u(x)|^2 + \sigma^2 \right) \, \mathrm{d}x$$



## A Few Classic Regularizers

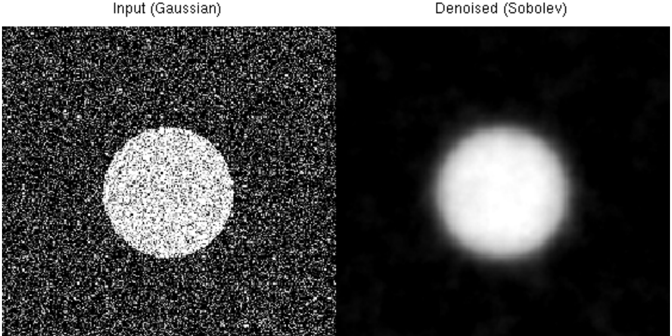
When modeling the restoration problem as the minimization of

$$E(u) = E_{data}(u, f) + \lambda\, E_{smoothness}(u),$$

several choices can be considered for the data term.

Choosing the (smooth and convex) Sobolev regularization tends to favor smoothness:

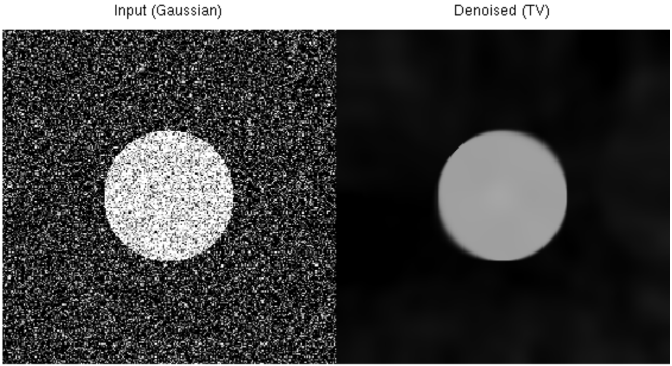
$$E_{smoothness}(u) = \int |\nabla u(x)|^2 \,\mathrm{d}x$$



# A Few Classic Regularizers

Choosing the (convex, yet non-smooth) Total variation (TV) tends to favor piecewise constantness:

$$E_{smoothness}(u) = \int \|\nabla u(x)\| \, \mathrm{d}x$$



# Overview

1. Inverse Problems and Variational Methods

2. **Variational Methods, Euler-Lagrange Equations and Diffusion**

3. Numerical Solving of the Restoration Problem

# Variational Methods

We have seen a few image processing problems which can be written as an optimization problem over a space of functions:

$$\min_{u:\,\Omega\subset\mathbb{R}^2\to\mathbb{R}} \quad E(u) := \int_\Omega \mathcal{L}(x, u(x), \nabla u(x))\,\mathrm{d}x$$

When $u$ is a vector, $E$ is a vectorial function and we know that a local minimum of such an optimization problem is characterized by $\nabla_u E = 0$.

How can we compute $\nabla_u E$ when $u$ is a function, and thus $E$ is a *functional* (function of a function) ?

I.e., how can we compute the *variations* of $E$ ? (*variational methods*)

# Functional Minimization





A functional is a mapping $E$ which assigns to each element of a vector-space (to each function $u$) an element from the underlying field (a number).

Let

$$E(u) = \int \mathcal{L}(u(x), u'(x))\, dx$$

be a functional, where $u' = \frac{du}{dx}$ is the derivative of the function $u$. (In physics $\mathcal{L}$ is called the Lagrange density).

Example: $\mathcal{L}(u(x), u'(x)) = \frac{1}{2}\big(u(x) - f(x)\big)^2 + \frac{\lambda}{2}|u'(x)|^2$.

Just as with real-valued functions defined on $\mathbb{R}^n$ the necessary condition for extremality of the functional $E$ states that the derivative with respect to $u$ must be 0.

Yet how does one define and compute the derivative of a functional $E(u)$ with respect to the function $u$?

## The Gâteaux Derivative

There are several ways to introduce functional derivatives. The following definition goes back to works of the French mathematician R. Gâteaux († 1914) which were published posthumously in 1919: http://www.numdam.org/article/BSMF_1919__47__47_1.pdf

The Gâteaux derivative extends the concept of directional derivative to infinite-dimensional spaces.

The derivative of the functional $E(u)$ in direction $h(x)$ is defined as:

$$\boxed{\left.\frac{dE(u)}{du}\right|_h = \lim_{\epsilon\to 0} \frac{E(u+\epsilon h) - E(u)}{\epsilon}}$$

As in finite dimensions, this directional derivative can be interpreted as the projection of the functional gradient on the respective direction. We can therefore write:

$$\left.\frac{dE(u)}{du}\right|_h = \left\langle \frac{dE(u)}{du}, h \right\rangle = \int \underbrace{\frac{dE(u)}{du}(x)}_{??} h(x)\, dx$$

## The Gâteaux Derivative

For functionals of the canonical form: $E(u) = \int \mathcal{L}(u, u')\, dx$ the Gâteaux derivative is given by

$$\left.\frac{dE(u)}{du}\right|_h = \lim_{\epsilon \to 0} \frac{1}{\epsilon}\left(E(u+\epsilon h) - E(u)\right)$$

$$= \lim_{\epsilon \to 0} \frac{1}{\epsilon} \int \left(\mathcal{L}(u+\epsilon h, u'+\epsilon h') - \mathcal{L}(u, u')\right) dx$$

$$= \lim_{\epsilon \to 0} \frac{1}{\epsilon} \int \left(\left(\mathcal{L}(u, u') + \frac{\partial \mathcal{L}}{\partial u}\epsilon h + \frac{\partial \mathcal{L}}{\partial u'}\epsilon h' + o(\epsilon^2)\right) - \mathcal{L}(u, u')\right) dx$$

$$= \int \left(\frac{\partial \mathcal{L}}{\partial u} h + \frac{\partial \mathcal{L}}{\partial u'} h'\right) dx$$

$$= \int \left(\frac{\partial \mathcal{L}}{\partial u} h - \frac{d}{dx}\frac{\partial \mathcal{L}}{\partial u'} h\right) dx \qquad \text{(partial int., } h = 0 \text{ on boundary)}$$

$$= \int \underbrace{\left(\frac{\partial \mathcal{L}}{\partial u} - \frac{d}{dx}\frac{\partial \mathcal{L}}{\partial u'}\right)}_{\frac{dE}{du}} h(x)\, dx.$$

# Euler-Lagrange Equation

Thus the derivative of the functional $E(u)$ in direction $h$ is:

$$\left.\frac{dE(u)}{du}\right|_h = \int \underbrace{\left(\frac{\partial \mathcal{L}}{\partial u} - \frac{d}{dx}\frac{\partial \mathcal{L}}{\partial u'}\right)}_{\frac{dE}{du}} h(x)\,dx.$$

As a necessary condition for minimality of the functional $E(u)$ the variation of $E$ in any direction $h(x)$ must vanish. Therefore at the extremum we have:

$$\boxed{\frac{dE}{du} = \frac{\partial \mathcal{L}}{\partial u} - \frac{d}{dx}\frac{\partial \mathcal{L}}{\partial u'} = 0}$$

This condition is called the Euler-Lagrange equation.

Example: For $\mathcal{L}(u, u') = \frac{1}{2}\big(u(x) - f(x)\big)^2 + \frac{\lambda}{2}|u'(x)|^2$, we get:

$$\frac{\partial \mathcal{L}}{\partial u} - \frac{d}{dx}\frac{\partial \mathcal{L}}{\partial u'} = \big(u(x) - f(x)\big) - \frac{d}{dx}\big(\lambda u'(x)\big) = u - f - \lambda u'' = 0$$

# Euler-Lagrange in 2D

Recall that in 1D we had:

$$\frac{dE}{du} = \frac{\partial \mathcal{L}}{\partial u} - \frac{d}{dx}\frac{\partial \mathcal{L}}{\partial u'}.$$

Extension to 2D is as follows:

$$\boxed{\frac{dE}{du} = \frac{\partial \mathcal{L}}{\partial u} - \operatorname{div}\frac{\partial \mathcal{L}}{\partial \nabla u}.}$$

For instance the derivative of the 2D denoising energy:

$$E(u) = \int (u-f)^2 dx + \lambda \int |\nabla u|^2\, dx$$

is given by

$$\frac{dE}{du} = 2(u-f) - 2\,\lambda\, \operatorname{div}\nabla u = 2\,(u - f - \lambda \Delta u)$$

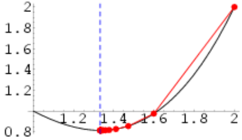
# Gradient Descent

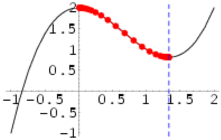
Gradient descent or steepest descent is a particular descent method where in each iteration one chooses the direction in which the energy decreases most. The direction of steepest descent is given by the negative energy gradient.

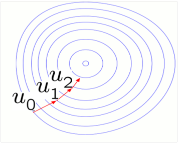
To minimize a real-valued function $f : \mathbb{R}^n \to \mathbb{R}$, the gradient descent for $f(u)$ is defined by the differential equation:

$$\begin{cases} u(0) = u_0 \\ \frac{du}{dt} = -\frac{df}{du}(u) \end{cases}$$

Discretization: $\quad u_{t+1} = u_t - \epsilon \frac{df}{du}(u_t), \qquad t = 0, 1, 2, \ldots.$



# Gradient Descent

For minimizing functionals $E(u)$, the gradient descent is done analogously.

For the functional $E(u) = \int \mathcal{L}(u, u')\, dx$, the gradient is given by:

$$\frac{dE}{du} = \frac{d\mathcal{L}}{du} - \frac{d}{dx}\frac{d\mathcal{L}}{du'}.$$

Therefore the gradient descent is given by:

$$\begin{cases} u(x,0) = u_0(x) \\ \dfrac{\partial u(x,t)}{\partial t} = -\dfrac{dE}{du} = -\dfrac{d\mathcal{L}}{du} + \dfrac{\partial}{\partial x}\dfrac{d\mathcal{L}}{du'}. \end{cases}$$

For $\mathcal{L}(u, u') = \frac{1}{2}\left(u - f\right)^2 + \frac{\lambda}{2}|u'|^2$, this means:

$$\frac{\partial u}{\partial t} = (f - u) + \lambda u'' = (f - u) + \lambda \Delta u.$$

If the gradient descent converges, i.e. $\partial_t u = -\frac{dE}{du} = 0$, then we have found a solution to the Euler-Lagrange equation.

# Overview

1. Inverse Problems and Variational Methods

2. Variational Methods, Euler-Lagrange Equations and Diffusion

3. **Numerical Solving of the Restoration Problem**

# Gradient Descent for the L2-TV (ROF) Model

Consider the generic L2-TV (ROF) restoration problem:

$$\min_{u\colon \Omega\subset\mathbb{R}^2\to\mathbb{R}} \int_\Omega \frac{1}{2}\left(u(x)-f(x)\right)^2 + \lambda\|\nabla u(x)\|\,\mathrm{d}x$$

Its first-order optimality condition is the Euler-Lagrange equation

$$u - f - \lambda \operatorname{div}\left(\frac{\nabla u}{\|\nabla u\|}\right) \quad \text{over } \Omega,$$

with Neumann or Dirichlet boundary conditions on $\partial\Omega$.

Starting from $u^{t=0} = f$, optimization can be carried out by gradient descent:

$$\partial_t = -u + f + \lambda \operatorname{div}\left(\frac{\nabla u}{\|\nabla u\|}\right).$$

# Main Issue in Gradient Descent for the L2-TV (ROF) Model





When implementing the gradient descent

$$\partial_t u = -u + f + \lambda \operatorname{div}\left(\frac{\nabla u}{\|\nabla u\|}\right).$$

one must be careful to avoid division by zero which occurs due to the factor $\|\nabla u\|$ (infinite diffusivity if there is no edge).

In practice, we need to smooth a bit this term using, e.g.,:

$$\frac{1}{\|\nabla u(x)\|} \approx \frac{1}{\|\nabla u(x)\|_\epsilon} := \frac{1}{\sqrt{\|\nabla u(x)\|^2 + \epsilon}}$$

with $\epsilon > 0$, small (e.g. $10^{-3}$)

## Explicit Time Gradient Descent

We can now discretize the gradient descent equation

$$\partial_t u = -u + f + \lambda \operatorname{div}\left(\frac{\nabla u}{\|\nabla u\|_\epsilon}\right)$$

wrt time $t$ using forward finite differences i.e.,

$$\partial_t u = \frac{u^{(t+1)} - u^{(t)}}{\delta_t},$$

with some fixed stepsize $\delta_t > 0$.

This yields the following algorithm:

$$\begin{aligned} u^{(0)} &= f \\ u^{(t+1)} &= u^{(t)} - \delta_t \left(u^{(t)} - f - \lambda \operatorname{div} \frac{\nabla u^{(t)}}{\|\nabla u^{(t)}\|_\epsilon}\right),\ t \in \{1, 2, \dots\} \end{aligned}$$

This works, but descent has to be slow (low $\delta_t$)

# Lagged Diffusivity (Implicit Time Gradient Descent)



To make things more stable, we usually prefer to freeze only the diffusivity during descent, i.e.:

$$u^{(0)} = f$$

$$u^{(t+1)} = u^{(t)} - \delta_t \left( u^{(t+1)} - f - \lambda \operatorname{div} \frac{\nabla u^{(t+1)}}{\|\nabla u^{(t)}\|_\epsilon} \right), \; t \in \{1, 2, \ldots\},$$

which requires a linear system to be solved at each update:

$$\left( (1 + \delta_t)\mathrm{id} - \delta_t \lambda \operatorname{div} \left( \frac{1}{\|\nabla u^{(t)}\|_\epsilon} \nabla \right) \right) u^{(t+1)} = u^{(t)} + \delta_t f$$

Typically much larger stepsizes are allowed, which makes things way faster and removes the need for tedious tuning (or linesearch).

# The Choice of the Stepsize Matters

Example with $\delta_t = 0.02$

# The Choice of the Stepsize Matters

Example with $\delta_t = 0.2$

# The Choice of the Stepsize Matters

Example with $\delta_t = 2$

# Gradient Descent Process for the Inpainting + Denoising Task