Mathematical Foundations of Data Sciences

Gabriel Peyré
CNRS & DMA, École Normale Supérieure

CHAPTER 8

Variational Priors and Regularization

September 9, 2026

Variational methods reconstruct images by balancing agreement with measured data against a penalty describing expected image structure. The choice of penalty controls the tradeoff between noise removal and preservation of sharp edges. Starting from Sobolev and total variation energies, we develop their discrete operators, derive the associated diffusion flows, and compare stopping a flow early with minimizing an energy that includes a data fidelity term.

8.1 Sobolev and Total Variation Priors

The simplest priors integrate local differential quantities over the image domain. They measure smoothness through seminorms on function spaces. We study two widely used examples: the Sobolev and total variation priors.

8.1.1 Continuous Priors

In this section we define priors for functions $f \in \mathrm{L}^2([0,1]^2)$ on a continuous domain. Section 8.1.2 develops their counterparts for discrete images $f \in \mathbb{R}^N$.

Sobolev prior. A prior energy assigns small values to images in the target class Θ and larger values, possibly $+\infty$, to images that depart from that model. The smoothness classes described in Section 5.2.1 motivate Sobolev priors. The simplest such energy is

$$J_{\mathrm{Sob}}(f) = \frac{1}{2}|f|_{H^1}^2 = \frac{1}{2}\int \|\nabla f(x)\|^2 \mathrm{d}x, \tag{8.1}$$

where ∇f is the distributional gradient and the integral is over $[0,1]^2$. The energy is one half of the squared H^1 seminorm and vanishes on constant functions. We set it to $+\infty$ when the gradient is not square integrable.

Total variation prior. To accommodate discontinuities in images, we use the total variation energy introduced in Section 5.2.3. Rudin, Osher, and Fatemi introduced its use in image denoising [1].

The total variation of a smooth image f is defined as

$$J_{\mathrm{TV}}(f) = \|f\|_{\mathrm{TV}} = \int \|\nabla_x f\| \mathrm{d}x. \tag{8.2}$$

This energy extends to functions of bounded variation $f \in \mathrm{BV}([0,1]^2)$, which may be discontinuous. This class includes indicator functions $f = 1_\Omega$ of sets Ω with finite perimeter $|\partial\Omega|$.

The total variation seminorm can also be computed using the coarea formula (5.12), which shows in particular that $\|1_\Omega\|_{\mathrm{TV}} = |\partial\Omega|$.

8.1.2 Discrete Priors

An acquisition device discretizes a continuous-domain image $f \in \mathrm{L}^2([0,1]^2)$ into a pixel array $f \in \mathbb{R}^N$. To process these data, we need priors defined on finite-dimensional image spaces.

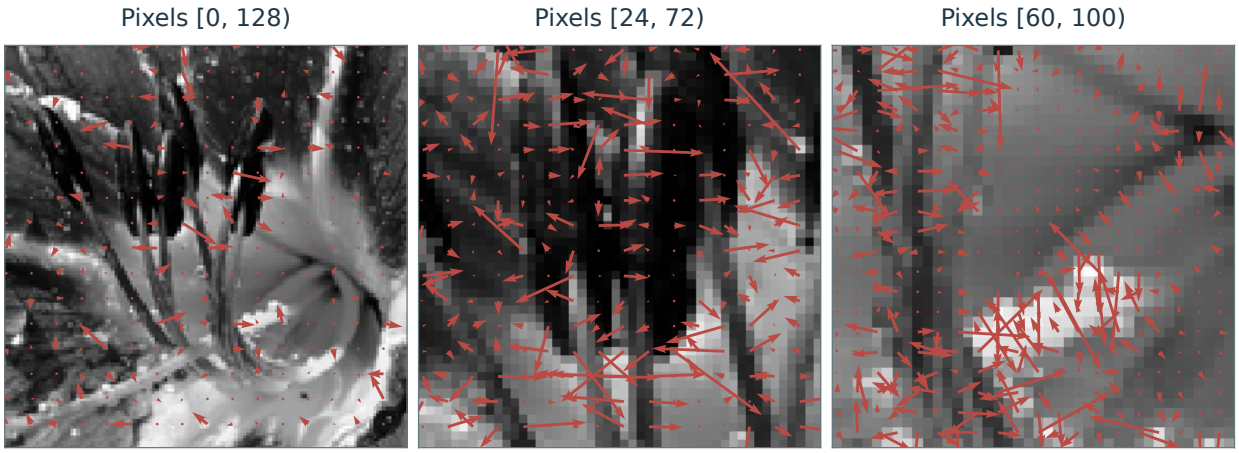

Figure 8.1. Discrete gradient vectors.

Discrete gradient. Discrete Sobolev and total variation priors use finite-difference approximations to derivatives. For example, the forward differences are

$$\delta_1 f_{n_1,n_2} = f_{n_1+1,n_2} - f_{n_1,n_2}$$
$$\delta_2 f_{n_1,n_2} = f_{n_1,n_2+1} - f_{n_1,n_2},$$

where the image has $n \times n$ pixels and $N = n^2$. Higher-order schemes give more accurate approximations for smooth functions. Boundary conditions require care. For simplicity, we use periodic boundary conditions, computing the indices $n_i + 1$ modulo n. Symmetric boundary conditions can also be used to avoid boundary artifacts.

With the grid-spacing factor absorbed into the energy scaling, the discrete gradient at each pixel is

$$\nabla f_n = (\delta_1 f_n, \delta_2 f_n) \in \mathbb{R}^2$$

and the gradient operator maps images to vector fields:

$$\nabla : \mathbb{R}^N \longrightarrow \mathbb{R}^{N\times 2}.$$

Figure 8.1 shows discrete gradient vectors approximating the local direction of steepest intensity increase. Figure 8.2 displays the signed gradient components and their Euclidean magnitude. A shared nonlinear display curve enhances weak red and blue components without changing the numerical gradient. Large magnitudes indicate rapid intensity variations, typically at edges or in textured regions.

Discrete divergence. One can also use backward differences,

$$\tilde{\delta}_1 f_{n_1,n_2} = f_{n_1,n_2} - f_{n_1-1,n_2}$$
$$\tilde{\delta}_2 f_{n_1,n_2} = f_{n_1,n_2} - f_{n_1,n_2-1}.$$

The adjoint relation between backward and forward differences is

$$\delta_i^* = -\tilde{\delta}_i,$$

which means that

$$\forall\, f, g \in \mathbb{R}^N, \quad \langle \delta_i f,\, g\rangle = -\langle f,\, \tilde{\delta}_i g\rangle,$$

This is the discrete counterpart of integration by parts,

$$\int_0^1 f' g = -\int_0^1 f g'$$

for smooth periodic functions on $[0, 1]$.

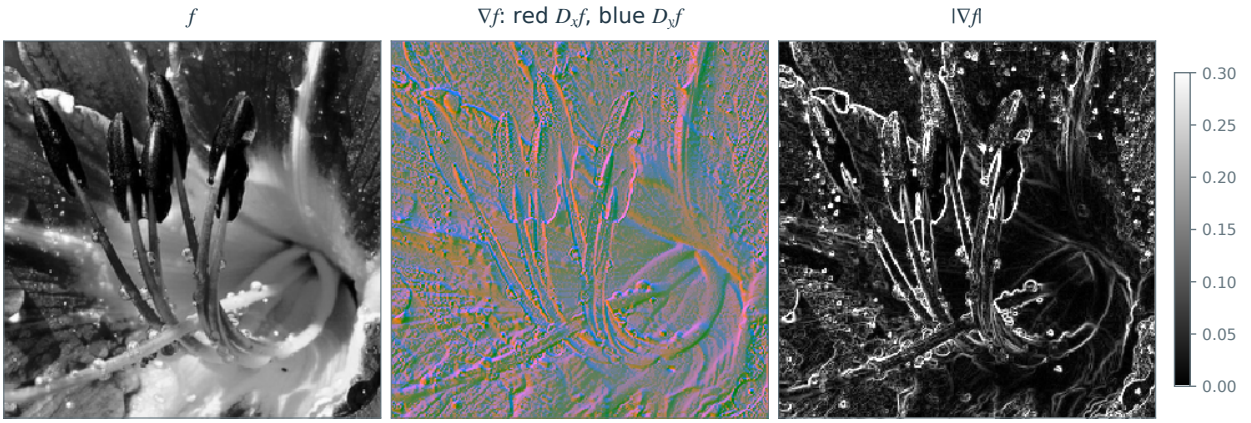

Figure 8.2. Discrete gradient of the flower image. Red and blue encode the signed horizontal and vertical components with enhanced display contrast; neutral gray represents zero. The last panel shows the Euclidean gradient magnitude.

We define the discrete divergence using backward differences:

$$\operatorname{div}(v)_n = \tilde{\delta}_1 v_{1,n} + \tilde{\delta}_2 v_{2,n},$$

This operator maps vector fields to images:

$$\operatorname{div} : \mathbb{R}^{N\times 2} \longrightarrow \mathbb{R}^N.$$

It is the negative adjoint of the gradient:

$$\operatorname{div} = -\nabla^*$$

which means that

$$\forall\, f \in \mathbb{R}^N,\ \forall\, v \in \mathbb{R}^{N\times 2},\quad \langle \nabla f,\, v\rangle_{\mathbb{R}^{N\times 2}} = -\langle f,\, \operatorname{div}(v)\rangle_{\mathbb{R}^N}$$

This identity is a discrete version of the divergence theorem.

Discrete Laplacian. The Laplacian is the divergence of the gradient:

$$\Delta f = \operatorname{div}(\nabla f),$$

This operator is negative semidefinite because $\Delta = -\nabla^*\nabla$.

With the discrete gradient and divergence defined above, the Laplacian is the local high-pass filter

$$\Delta f_n = \sum_{p\in V_4(n)} f_p - 4 f_n, \tag{8.3}$$

which, after scaling by the grid spacing, approximates the continuous Laplacian:

$$\frac{\partial^2 f}{\partial x_1^2}(x) + \frac{\partial^2 f}{\partial x_2^2}(x) \approx n^2 \Delta f_k \quad \text{for} \quad x = k/n.$$

Laplacian operators thus act as filters. With the Fourier convention $\hat{f}(\omega) = \int f(x) e^{-\mathrm{i}\langle x,\omega\rangle}\,\mathrm{d}x$, the continuous Laplacian is diagonal in the Fourier domain:

$$g = \Delta f \quad \Longrightarrow \quad \hat{g}(\omega) = -\|\omega\|^2 \hat{f}(\omega)$$

while the discrete Laplacian (8.3) has Fourier representation

$$g = \Delta f \quad \Longrightarrow \quad \hat{g}_\omega = -\rho_\omega^2 \hat{f}_\omega \quad \text{where} \quad \rho_\omega^2 = 4\sin\left(\frac{\pi}{n}\omega_1\right)^2 + 4\sin\left(\frac{\pi}{n}\omega_2\right)^2. \tag{8.4}$$

Discrete energies. The discrete Sobolev energy is one half of the squared ℓ^2 norm of the gradient field:

$$J_{\mathrm{Sob}}(f) = \frac{1}{2}\sum_n \left((\delta_1 f_n)^2 + (\delta_2 f_n)^2\right) = \frac{1}{2}\|\nabla f\|^2. \tag{8.5}$$

The isotropic discrete TV energy instead sums the Euclidean magnitudes of the gradient vectors:

$$J_{\mathrm{TV}}(f) = \sum_n \sqrt{(\delta_1 f_n)^2 + (\delta_2 f_n)^2} = \|\nabla f\|_1 \tag{8.6}$$

Here the notation $\|v\|_1$ for a vector field $v \in \mathbb{R}^{N\times 2}$ means

$$\|v\|_1 = \sum_n \|v_n\| \tag{8.7}$$

where $v_n \in \mathbb{R}^2$.

8.2 PDE and Energy Minimization

Minimizing a smooth prior by gradient descent produces an image-smoothing flow.

8.2.1 General Flows

The gradient of a differentiable prior $J : \mathbb{R}^N \to \mathbb{R}$ is the vector $\operatorname{grad} J(f)$ that describes its local first-order variation:

$$J(f+\varepsilon) = J(f) + \langle \varepsilon,\, \operatorname{grad} J(f)\rangle + o(\|\varepsilon\|).$$

Gradient descent seeks to decrease a smooth discrete energy by iterating

$$f^{(k+1)} = f^{(k)} - \tau \operatorname{grad} J(f^{(k)}), \tag{8.8}$$

where the step size τ is chosen to ensure descent. The quadratic and smoothed TV energies below admit explicit sufficient bounds on τ.

To pass to continuous time, associate iteration k with time $t = k\tau$ and let τ tend to zero. The limiting flow

$$t > 0 \longmapsto f_t \in \mathbb{R}^N$$

satisfies the following differential equation, an ODE for a finite-dimensional image and a PDE when space is continuous:

$$\frac{\partial f_t}{\partial t} = -\operatorname{grad} J(f_t) \quad \text{and} \quad f_0 = f. \tag{8.9}$$

Gradient descent is an explicit time discretization of this differential equation at times $t_k = k\tau$.

8.2.2 Heat Flow

Heat flow results from applying (8.9) to the Sobolev energy $J_{\mathrm{Sob}}(f)$, defined in (8.1) for functions and in (8.5) for discrete images.

Expanding the quadratic energy gives

$$J(f+\varepsilon) = \frac{1}{2}\|\nabla f + \nabla\varepsilon\|^2 = J(f) - \langle \Delta f,\, \varepsilon\rangle + \frac{1}{2}\|\nabla\varepsilon\|^2,$$

so that

$$\operatorname{grad} J_{\mathrm{Sob}}(f) = -\Delta f.$$

Figure 8.4, left, shows an image Laplacian. Its magnitude is typically large near edges, with either sign.

The heat flow is thus

$$\frac{\partial f_t}{\partial t}(x) = -(\operatorname{grad} J(f_t))(x) = \Delta f_t(x) \quad \text{and} \quad f_0 = f. \tag{8.10}$$

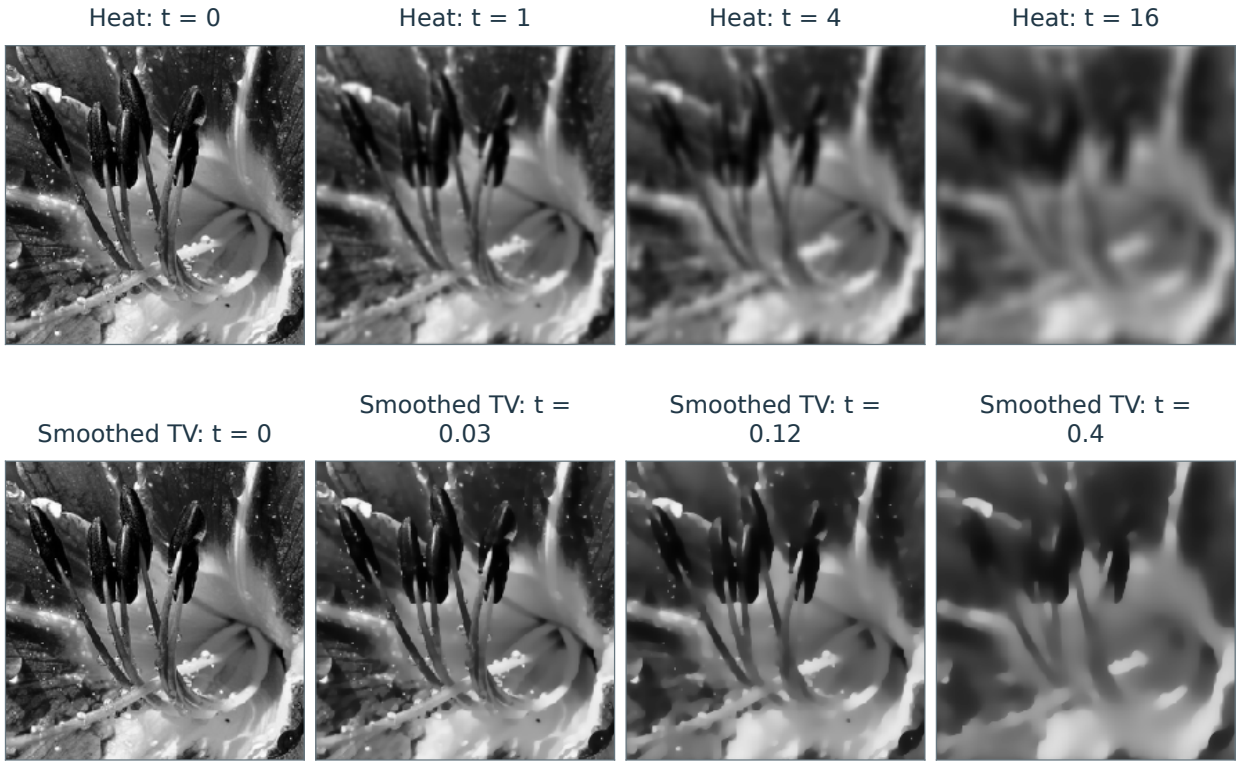

Figure 8.3. Heat flow (top) and TV flow (bottom): images f_t at increasing times t.

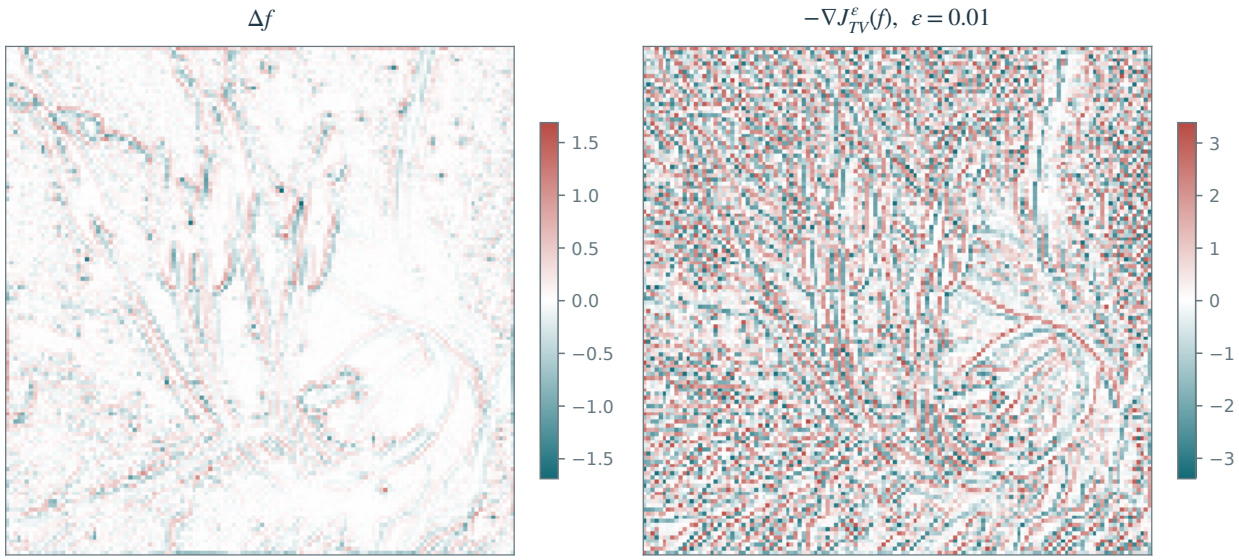

Figure 8.4. Discrete Laplacian (left) and negative discrete TV gradient (right).

Continuous in space. For continuous images on the unbounded domain $\mathbb{R}^2$, the PDE (8.10) has an explicit solution: convolution with a Gaussian kernel whose variance grows with time

$$f_t = f \star h_t \quad \text{where} \quad h_t(x) = \frac{1}{4\pi t} e^{-\frac{\|x\|^2}{4t}}. \tag{8.11}$$

Convolution with this widening kernel progressively smooths the image.

Discrete in space. The gradient flow of the discrete Sobolev energy (8.5) is

$$\frac{\partial f_{n,t}}{\partial t} = -(\operatorname{grad} J(f_t))_n = (\Delta f_t)_n.$$

Discretizing time as in (8.8) gives the fully discrete iteration

$$f_n^{(k+1)} = f_n^{(k)} + \tau\Big(\sum_{p \in V_4(n)} f_p^{(k)} - 4 f_n^{(k)} \Big) = (f^{(k)} \star h)_n$$

where $V_4(n)$ denotes the four neighbors of pixel n. Each iteration convolves the current image with the same filter:

$$f^{(k)} = f \star h \star \ldots \star h = f \star^k h.$$

The filter has coefficients $h_0 = 1 - 4\tau$ and $h_{\pm e_1} = h_{\pm e_2} = \tau$; all other entries vanish.

This iteration is stable and converges if $0 < \tau < 1/4$.

8.2.3 Total Variation Flows

Total variation gradient. The total variation energy J_{TV} is nonsmooth, both in its continuous form (8.2) and in its discrete form (8.6). In the discrete case, J_{TV} is nondifferentiable at any image f with a pixel n where $\nabla f_n = 0$.

If $\nabla f_n \neq 0$ at every pixel, the gradient is

$$(\operatorname{grad} J(f))_n = -\operatorname{div}\left(\frac{\nabla f}{\|\nabla f\|} \right)_n$$

The displayed quotient is undefined where the spatial gradient vanishes. Figure 8.4, right, shows the negative TV gradient; normalization by small gradient magnitudes makes it sensitive to perturbations in smooth regions.

Nondifferentiability prevents the use of classical gradient descent at such images. A TV flow can nevertheless be defined through the subgradient inclusion $\partial_t f_t \in -\partial J_{\mathrm{TV}}(f_t)$. Here we first study a smooth approximation.

Regularized total variation. To obtain a differentiable energy, we smooth the TV prior as follows:

$$J_{\mathrm{TV}}^{\varepsilon}(f) = \sum_n \sqrt{\varepsilon^2 + \|\nabla f_n\|^2} \tag{8.12}$$

where $\varepsilon > 0$ controls the smoothing. Figure 8.5 illustrates the corresponding smoothing of the absolute value function.

The smoothed TV energy has gradient

$$\operatorname{grad} J_{\mathrm{TV}}^{\varepsilon}(f) = -\operatorname{div}\left(\frac{\nabla f}{\sqrt{\varepsilon^2 + \|\nabla f\|^2}} \right). \tag{8.13}$$

This smoothing interpolates between TV and a rescaled Sobolev energy. For fixed f,

$$\operatorname{grad} J_{\mathrm{TV}}^{\varepsilon}(f) = -\Delta f/\varepsilon + O(\varepsilon^{-3}) \quad \text{when} \quad \varepsilon \to +\infty.$$

Figure 8.6 displays the regularized spatial-gradient magnitude for several smoothing parameters.

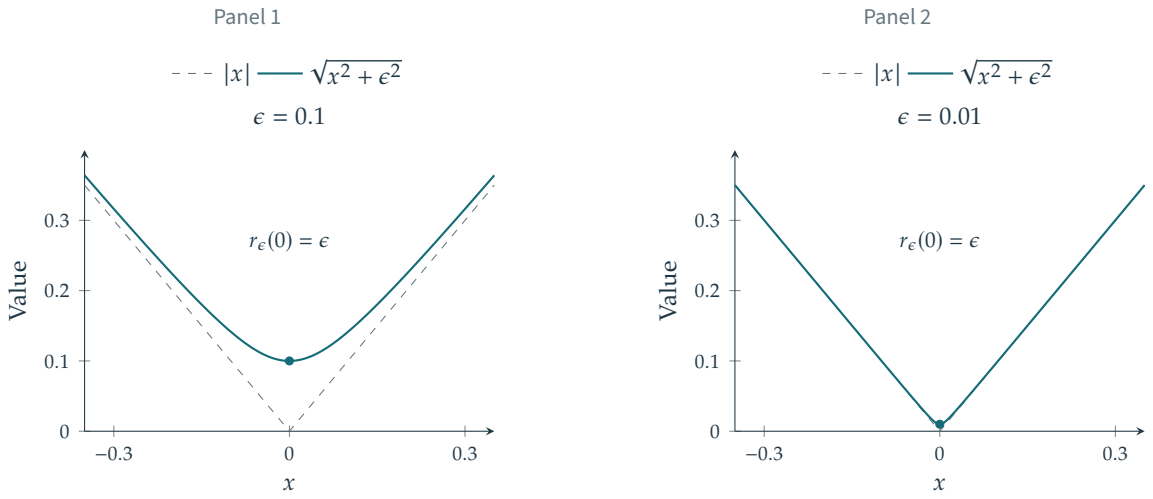

Figure 8.5. Regularized absolute value $x \mapsto \sqrt{x^2 + \varepsilon^2}$.

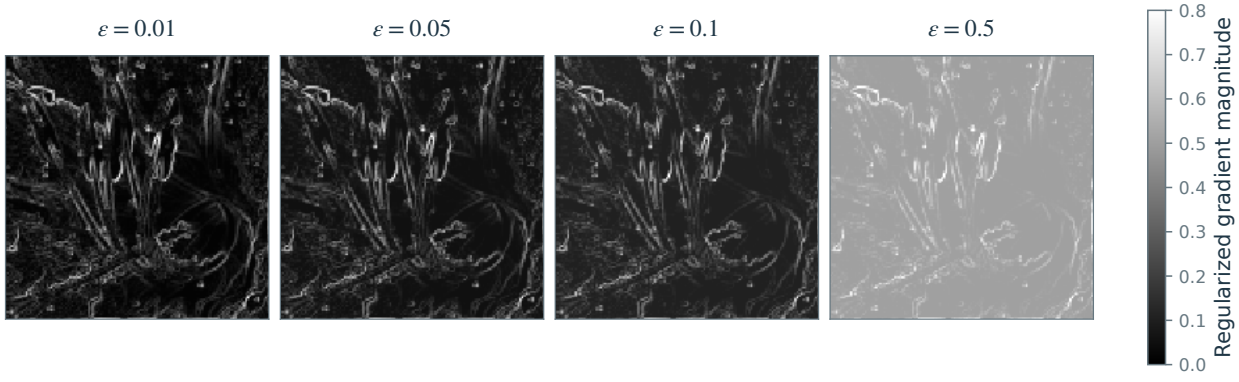

Figure 8.6. Regularized gradient norm $\sqrt{\|\nabla f(x)\|^2 + \varepsilon^2}$.

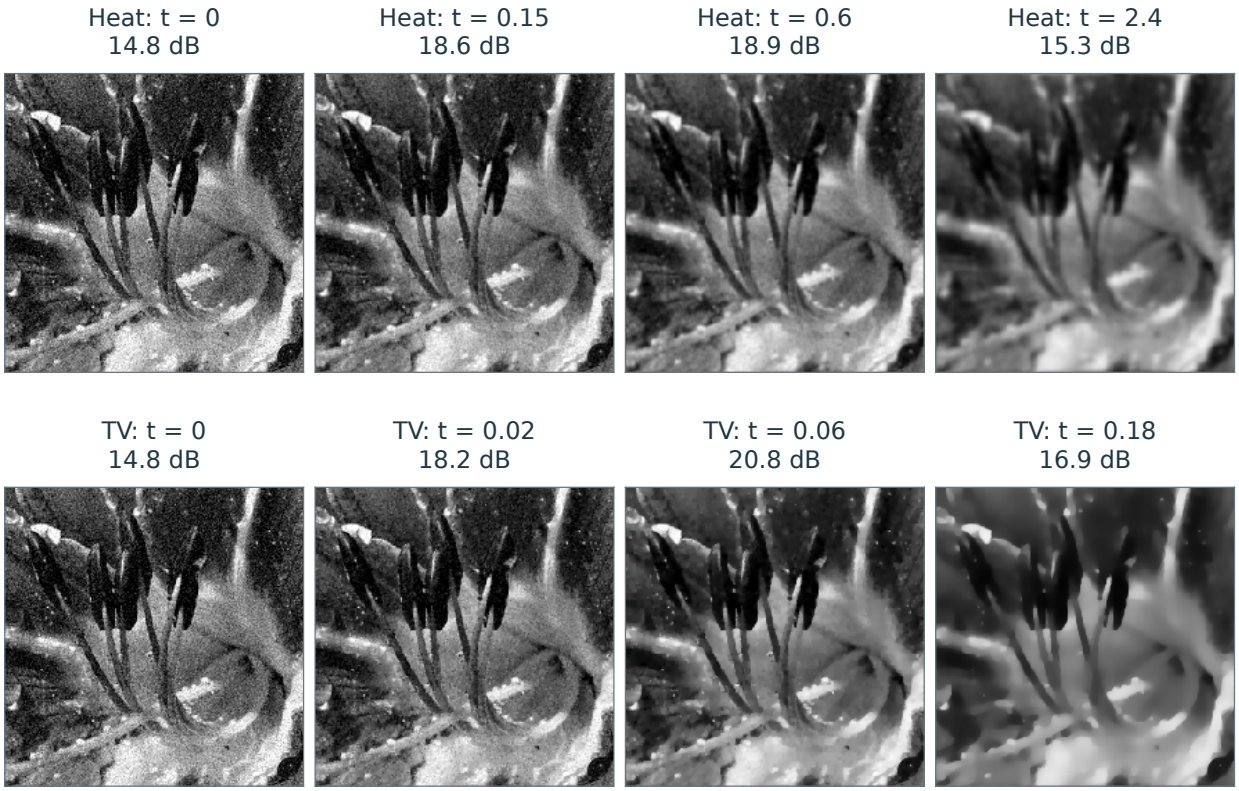

Figure 8.7. Denoised images f_t at several times t: Sobolev flow (top) and TV flow (bottom).

Regularized total variation flow. The smoothed total variation flow is then defined as

$$\frac{\partial f_t}{\partial t} = \operatorname{div}\left(\frac{\nabla f_t}{\sqrt{\varepsilon^2 + \|\nabla f_t\|^2}}\right). \tag{8.14}$$

Choosing a small ε makes the flow approximate total variation minimization more closely, but requires smaller time steps in explicit discretizations.

In practice, gradient descent (8.8) discretizes this flow in time. For the smoothed total variation flow to converge, a sufficient condition is $0 < \tau < \varepsilon/4$. Thus, a closer approximation to the TV energy requires smaller time steps and more iterations.

Figure 8.3 compares heat flow with smoothed TV flow for a small ε. TV flow preserves edges better than heat diffusion, consistent with the ability of the TV energy to represent sharp transitions.

8.2.4 PDE Flows for Denoising

PDE flows can be used to remove noise from an observation $f = f_0 + w$. As detailed in Section 7.1.2, a simple noise model assumes that each pixel is corrupted by Gaussian noise $w_n \sim \mathcal{N}(0, \sigma^2)$, and that these perturbations are independent (white noise).

To denoise, we initialize the PDE flow at the observed image f at time $t = 0$:

$$\frac{\partial f_t}{\partial t} = -\operatorname{grad}_{f_t} J \quad \text{and} \quad f_{t=0} = f.$$

Stopping the flow defines an estimator $\tilde{f} = f_{t_0}$, whose quality depends on the stopping time t_0. Figure 8.7 illustrates denoising by Sobolev and TV flows.

On a connected periodic grid, f_t converges to a constant image when $t \to +\infty$, so the choice of t_0 balances noise removal against excessive smoothing of image edges. Selecting this stopping time is often difficult. In simulations, if one has access to the clean image f_0, one can monitor the denoising error $\|f_0 - f_t\|$ and choose the $t = t_0$ that minimizes this error. Figure 8.8, top row, shows reconstructions obtained with such an oracle choice of stopping time.

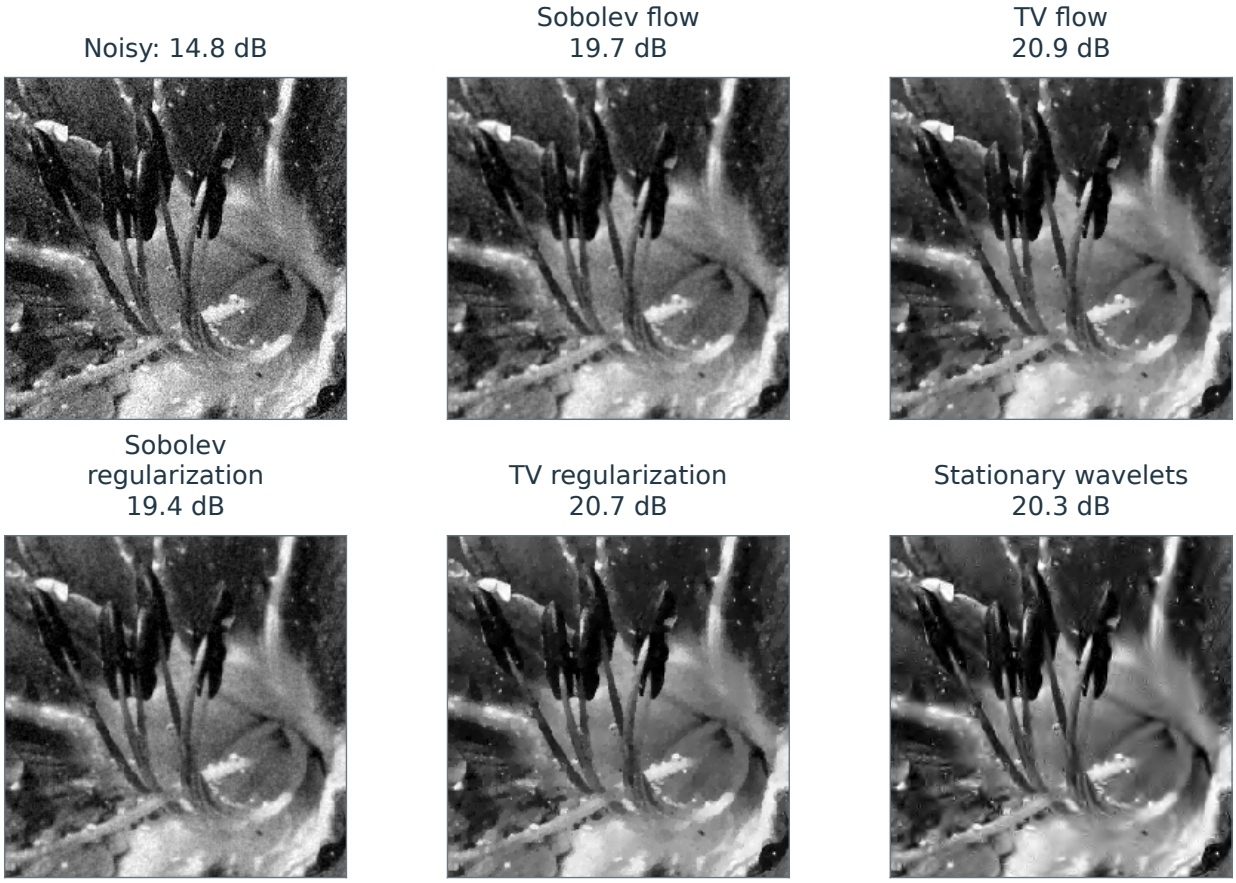

Figure 8.8. Denoising using PDE flows and regularization.

8.3 Regularization for Denoising

The flow-based approach in Section 8.2.4 defines the estimator by choosing a stopping time. Alternatively, we can define it as the minimizer of an objective that balances data fidelity and a prior. This formulation also accommodates nonsmooth priors such as total variation J_{TV} and sparsity J_1. Chapter 9 extends it to general inverse problems.

8.3.1 Regularization

For a noisy image $f = f_0 + w$ with N pixels and a prior J, consider

$$f_\lambda^\star \in \operatorname*{argmin}_{g \in \mathbb{R}^N} \frac{1}{2}\|f - g\|^2 + \lambda J(g), \tag{8.15}$$

where the regularization parameter $\lambda > 0$ balances the data fidelity term $\|f - g\|^2$ against the penalty $J(g)$.

If a clean reference image f_0 is available, one can choose the parameter by minimizing the denoising error $\|f_\lambda^\star - f_0\|$. Such a reference is rarely available in practice, so parameter selection must account for the noise level and the regularity of the unknown image f_0.

For a proper lower semicontinuous convex prior J, the quadratic fidelity makes the objective strongly convex, so it has a unique minimizer. For nonconvex priors, minimizers need not be unique; properness, lower semicontinuity, and a lower bound for J suffice for existence.

The resulting estimator is

$$\tilde{f} = f_\lambda^\star$$

with λ chosen according to the noise level and prior.

For a differentiable convex prior J, gradient descent for (8.15) takes the form

$$f^{(k+1)} = f^{(k)} - \tau \left(f^{(k)} - f + \lambda \operatorname{grad} J(f^{(k)}) \right) \tag{8.16}$$

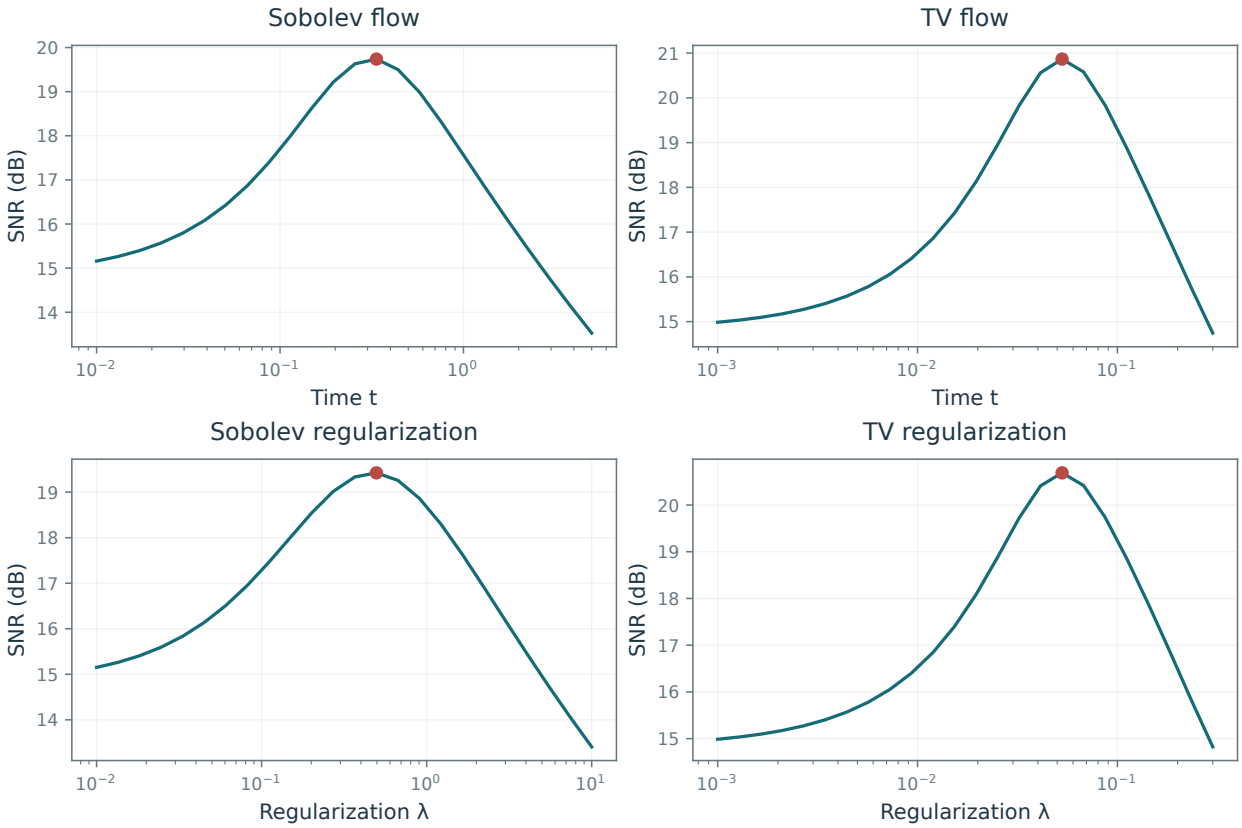

Figure 8.9. SNR as a function of time t for flows (top) and λ for regularization (bottom).

where $\tau > 0$ is chosen according to the smoothness of the objective. The iteration adds a data fidelity force to the time-discretized flow (8.8), counteracting the prior's tendency to smooth the image toward a constant.

Total variation J_{TV} and the sparsity prior J_1 are nondifferentiable; the ideal sparsity prior J_0 is also nonconvex. Classical gradient descent therefore cannot be applied directly to (8.15). Computing $f_\lambda^\star$ then requires smoothing the prior or using algorithms adapted to nonsmooth objectives, as developed later in the book.

8.3.2 Sobolev Regularization

The discrete Sobolev prior defined in (8.5) is differentiable, and the gradient descent (8.16) reads

$$f^{(k+1)} = (1-\tau)f^{(k)} + \tau f + \tau\lambda\Delta f^{(k)}.$$

Since $J(f) = \frac{1}{2}\|\nabla f\|^2$ is quadratic, one can also solve the associated linear system by conjugate gradients.

The solution $f_\lambda^\star$ has the explicit linear-system representation

$$f_\lambda^\star = (\mathrm{Id}_N - \lambda\Delta)^{-1} f,$$

Thus, for this prior, the estimator defined by (8.15) depends linearly on the observations f.

If the differential operators are computed with periodic boundary conditions, this linear system can be solved exactly in the Fourier domain

$$(\hat{f}_\lambda^\star)_\omega = \frac{1}{1+\lambda\rho_\omega^2}\hat{f}_\omega \tag{8.17}$$

where ρ_ω is determined by the discrete Laplacian; see (8.4).

Equation (8.17) shows that denoising using Sobolev regularization corresponds to low-pass filtering with strength controlled by λ. Compare this with the solution (8.11) of the heat equation, which uses a Gaussian low-pass kernel with variance $2t$ in each coordinate.

Sobolev denoising belongs to the class of linear estimators studied in Section 7.2. Choosing λ is therefore a filter-selection problem, as in Section 7.2.2, restricted here to a one-parameter family.

8.3.3 TV Regularization

The total variation prior J_{TV} in (8.6) is nondifferentiable. We can either smooth it or use an algorithm designed for nonsmooth optimization.

The approximation $J_{\mathrm{TV}}^{\varepsilon}(g)$ defined in (8.12) is differentiable in g. Substituting its gradient (8.13) into (8.16) gives

$$f^{(k+1)} = (1-\tau) f^{(k)} + \tau f + \lambda \tau \operatorname{div}\left(\frac{\nabla f^{(k)}}{\sqrt{\varepsilon^2 + \|\nabla f^{(k)}\|^2}} \right). \tag{8.18}$$

For $0 < \tau < 2/(1 + 8\lambda/\varepsilon)$, these iterates converge to the unique minimizer of (8.15) with $J = J_{\mathrm{TV}}^{\varepsilon}$.

Section 19.5.1 presents an algorithm for the original nonsmooth TV penalty.

References

[1] L. I. Rudin, S. Osher, and E. Fatemi. Nonlinear total variation based noise removal algorithms. *Phys. D*, 60(1-4):259–268, 1992.