Chapter 1

Image Restoration

ch,restore

Contents

s,res,intro

1.1 Introduction (s,res,intro)

Although this book emphasizes tomographic **image reconstruction** problems, many of the concepts apply to the related inverse problem known as **image restoration**. The physical models used in basic image restoration problems are often simpler than those of realistic tomographic reconstruction problems, so image restoration problems provide a convenient framework in which to describe many of the principles of **image recovery** problems[1] in general, both restoration and reconstruction problems[2]. This *chapter* uses the context of image restoration problems to provide an overview of the concepts that will follow later in the book. By no means is this an exhaustive treatment of image restoration; the focus is on principles that are common to both restoration and reconstruction. Readers familiar with statistical image restoration methods could simply skim this chapter for the notation conventions. For further reading, see [2]–[6].

In 2D image restoration problems, we are given a blurry, noisy image

e,res,gmn

$$\{g[m,n] \in \mathbb{R} : m = 0, \ldots, M-1, \; n = 0, \ldots, N-1\} \tag{1.1.1}$$

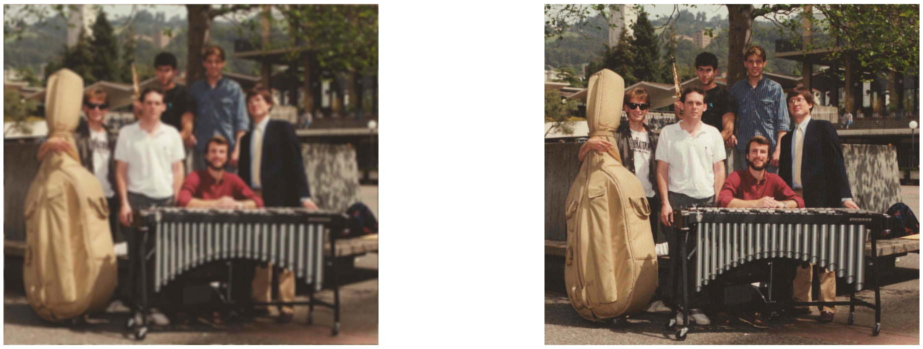
of some object or scene, recorded, for example, by a camera with a digital detector such as a **CCD array** [7], [8]. Restoration of astronomical images from the Hubble Space Telescope is a famous example of such a problem [9]. Our goal is to use the measured image to form an estimate of the underlying "true" object, *i.e.*, to eliminate or reduce the blur and the noise. We focus on 2D problems, but the concepts generalize readily to 3D restoration problems like confocal microscopy [10] and to problems with multiple views of the same object [11]–[14].

Fig. 1.1.1 illustrates image restoration using a method described in §1.8.2.

To proceed, we must define the restoration goal more precisely. An ambitious goal would be to try to reconstruct the image that would be recorded by an imaging device with perfect spatial resolution, sufficiently large field of view, arbitrary dynamic range, and no measurement errors (noiseless). Practical formulations are usually less ambitious, as described next.

[1] There is not universal terminology for image recovery problems. Typically **image reconstruction** means forming an image from measured data that is not interpretable directly as an image, such as a **sinogram** in tomography. In contrast, in **image restoration** problems one begins with an image (usually noisy and blurry) and tries to improve it. A special case is **image denoising** where one tries to reduce noise without considering blur. We use **image recovery** to encompass all such problems.

[2] In the context of medical imaging, iterative methods for image restoration were applied as far back as 1967 [1].



fig_res_ex2b Figure 1.1.1: The left image was degraded by a 11×11 uniform blur $b[m,n]$ and additive gaussian noise corresponding to 60 dB blurred signal-to-noise ratio (**BSNR**). The right image illustrates deblurring using the method in §1.8.2. fig_res_ex2a

1.2 Conventional discrete measurement model

The simplest formulations of image restoration problems assume that the object to be recovered is also a discrete-space array with the same domain as $g[m,n]$. In other words, we wish to estimate the **latent image** (unknown object)

e,res,domain

$$\{f[m,n] \in \mathbb{R} : m = 0, \ldots, M-1,\ n = 0, \ldots, N-1\} \tag{1.2.1}$$

from the measurements $\{g[m,n]\}$. The first step in formulating any inverse problem is to define a **system model** that relates the unknown quantities to the observed measurements. In the signal processing literature, this is known as the **input-output relationship** of the system. In the inverse problems field, this is called the **forward model** for the problem. Several of the subsequent chapters focus on system models for tomographic systems.

For image restoration, the simplest model is to assume that the imaging system is **linear** and **shift-invariant**. The input-output relationship of any linear shift-invariant (LSI) system can be represented by a **convolution** operation. For a discrete-space LSI system, the convolution sum is[3]

e,res,conv

$$\bar{g}[m,n] = b[m,n] ** f[m,n] \triangleq \sum_{k=0}^{M-1} \sum_{l=0}^{N-1} b[m-k, n-l]\, f[k,l], \tag{1.2.2}$$

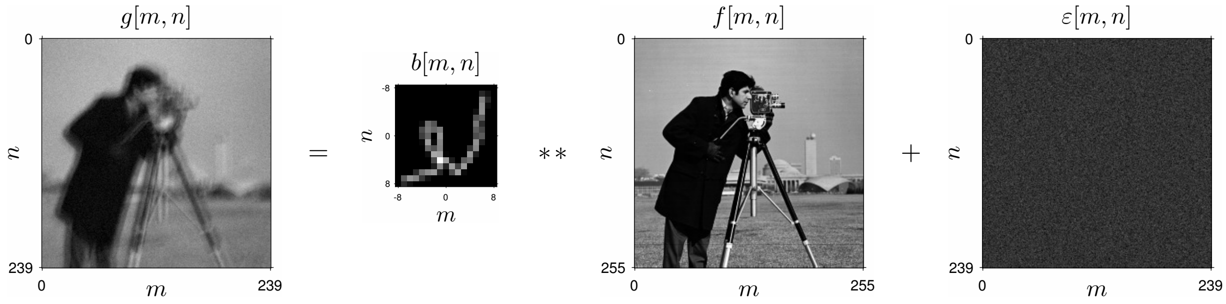
where $b[m,n]$ denotes the **impulse response** of the system, or specifically the **point spread function** (**PSF**) in the context of imaging problems. Typically $b[m,n]$ causes *blur*.

The convolution model (1.2.2) by itself is incomplete because any real imaging system has measurement errors, known as "**noise**." (This terminology comes from the influence of audio signals in the signal processing field). Several subsequent chapters describe statistical models for tomographic measurements. For image restoration, the simplest model assumes that the measurement noise $\varepsilon[m,n]$ is additive:

e,res,g=h*f+e

$$g[m,n] = b[m,n] ** f[m,n] + \varepsilon[m,n], \qquad m = 0, \ldots, M-1,\ n = 0, \ldots, N-1. \tag{1.2.3}$$

Often it is assumed that the noise is zero mean and has a gaussian distribution. Fig. 1.2 illustrates this model.



fig_res_gbfv

Figure 1.2.1: Graphical illustration of the imaging model (1.2.3) for the **extended end conditions** of §1.4.1.2.

With the statistical model (1.2.3) in hand, we can state that the goal of image restoration is to recover $\{f[m,n]\}$ from $\{g[m,n]\}$, using knowledge of the PSF $b[m,n]$ and a statistical model for the noise $\varepsilon[m,n]$. A challenging variation of this problem is **blind restoration** in which the PSF $b[m,n]$ is also unknown or partially unknown, *e.g.*,

[3] The notation $\bar{g}[m,n] = (b ** f)[m,n]$ would be more precise. The standard definition of the 2D convolution sum is $(b ** f)[m,n] = \sum_{k=-\infty}^{\infty} \sum_{l=-\infty}^{\infty} b[m-k, n-l]\, f[k,l]$. For now, we treat $f[\cdot,\cdot]$ as be zero for arguments outside its domain (1.2.1). We revisit this assumption in §1.4.

[15]–[37]. This problem is particularly interesting when given multiple images with different blurs [12], [28], [38]–[43].

Although (1.2.3) is the classical model for image restoration that has been the focus of numerous publications, there are very few, if any, realistic problems that exactly match the underlying assumptions! The unknown object is rarely a discrete-space function, imaging systems are rarely perfectly shift invariant (due to problems like optical aberrations), and in optical imaging the noise is rarely exactly gaussian due to the quantum effects of photons. Furthermore, digital detectors always quantize the measurements. There are also subtle effects at the borders of the **field of view** that require consideration.

The mismatch between the classical model and physical reality is not unique to image restoration problems; many tomography papers have also been based on over-simplified models. Simplified models are certainly very important for developing intuition and, in many cases, for finding fast (albeit approximate) algorithms. But one of the goals of subsequent chapters is to describe reasonably complete models for tomographic systems because one of the most important benefits of statistical methods for image reconstruction is that they can incorporate realistic models, whereas analytical reconstruction methods usually are limited to relatively simplified models.

As a preview of the types of system models that are described in subsequent chapters, we consider next a somewhat more realistic model for many image restoration problems, and attempt to relate it to (1.2.3).

s,res,cd

1.3 Continuous-discrete model (s,res,cd)

Although no physical system can be perfectly linear over an arbitrary dynamic range, practical systems are usually designed to operate within the linear range of the system components. So the linearity assumption that underlies (1.2.2) is often reasonable. However, in most image restoration problems the (unknown) “true object” is a function of continuous arguments, called a **continuous-space function**, *e.g.*, $\{f(x,y) : x, y \in \mathbb{R}\}$, so using a discrete-space object and PSF in (1.2.2) is a simplification. Compared to (1.2.2), a more realistic measurement model for image restoration problems is the following [44]:

e,res,cd,model

$$g[m,n] = \bar{g}[m,n] + \varepsilon[m,n], \quad \bar{g}[m,n] = \iint b(m,n;x,y)\, f(x,y)\, \mathrm{d}x\, \mathrm{d}y, \tag{1.3.1}$$

where $b(m,n;x,y)$ denotes the contribution that an impulse object located at (x,y) would make to the expected value of $g[m,n]$. The function $b(m,n;x,y)$, combined with the assumption of linearity, completely describes the non-statistical aspects of the imaging system, so we refer to $b(m,n;x,y)$ as the **system model**.

The model (1.3.1) is known as a **continuous to discrete mapping**. With this model, we could pose the problem as one of estimating f given $\{g[m,n]\}$. This problem is the epitome of under-determined; there are uncountably many f that agree exactly with the measurements $g[m,n]$, even for noiseless data where $\varepsilon[m,n] = 0$.

1.3.1 Ill-posed problems

Trying to reconstruct f from $\{g[m,n]\}$ is an example of an **ill-posed problem**. Problems are called **well-posed** in the sense of Hadamard [45, p. 63] when a solution exists that is unique and stable (continuous) with respect to the data. Otherwise a problem is called ill-posed.

In imaging problems with a underlying continuous-space object f but a finite number of measurements, non-uniqueness is the principal challenge. Clearly we must impose some type of prior knowledge (or prior assumptions) about f to proceed.

Most methods in the literature avoid the complications of the continuous-discrete model (1.3.1) by adopting either a continuous-continuous formulation, or, more often, the discrete-discrete formulation (1.2.2). An example of a continuous-continuous model is

e,res,cc,g=b*f+e

$$g(x,y) = b(x,y) ** f(x,y) + \varepsilon(x,y), \tag{1.3.2}$$

where $b(x,y)$ denotes a continuous-space PSF and $\varepsilon(x,y)$ denotes a finite-power random noise process. One can derive estimators $\hat{f}(x,y)$ in terms of the “given” continuous-space image $g(x,y)$ in this framework. One simple example is the continuous-space **Wiener filter** (*cf.* the discrete-space version in (1.7.11)). Another broader family of methods uses partial differential equation (**PDE**) methods to address (1.3.2) [46]. Of course in practice we never have a continuous-space $g(x,y)$ available, so one must somehow discretize any such solution expressions to implement a practical estimator in terms of $g[m,n]$.

Although subsequent sections focus on the discrete-discrete formulation (1.2.3), for completeness we mention one particular approach that is formulated directly from the continuous-discrete model (1.3.1).

1.3.2 Minimum norm estimate

Even if there were no noise in (1.3.1), *i.e.*, if $\varepsilon[m,n] = 0$, there would still be many images f that satisfy exactly the equalities in (1.3.1). One way to pick a solution from this infinite set is to choose the solution with minimum

$\mathcal{L}_2$ **norm** (*i.e.*, **minimum energy** in signal processing terminology). By the **projection theorem** (an important result from functional analysis [47, p. 51]), a unique solution with minimum norm exists. It can be expressed as

e,res,cd,minnorm

$$\hat{f} = \operatorname*{arg\,min}_{f \in \mathcal{C}(g)} \iint |f(x,y)|^2 \,\mathrm{d}x \,\mathrm{d}y, \tag{1.3.3}$$

where $\mathcal{C}(g)$ denotes the following **convex set** (see §27.9.1):

e,res,C(g)

$$\mathcal{C}(g) \triangleq \left\{ f \,:\, \iint b(m,n;x,y)\, f(x,y) \,\mathrm{d}x \,\mathrm{d}y = g[m,n], \quad \forall m,n \right\}. \tag{1.3.4}$$

One can calculate the minimum norm solution using a certain **singular value decomposition** (**SVD**) [48]. The resulting solution has the form

e,res,mnls

$$\hat{f}(x,y) = \sum_{m,n} c_{m,n}\; b(m,n;x,y) \tag{1.3.5}$$

for some coefficients $\{c_{m,n}\}$ that depend on the data $\{g[m,n]\}$ and on the system model $b(\cdot)$. This minimum-norm solution is convenient for analysis, but rarely is there any physical reason why real objects should be of minimum norm, so this criterion for picking one of many solutions is just as *ad hoc* as any alternative. (In fact, one could replace the usual norm in (1.3.3) with many alternative norms (or semi-norms) [49]–[51].) Furthermore, enforcing strict data consistency in (1.3.4) essentially means that noise is ignored.

The subject of **regularization** is closely related to this problem of choosing from an infinite collection of solutions, and is a major focus of Chapter 2 and other portions of several chapters of this book.

1.3.3 Object models

The minimum norm approach is one way to impose (fairly weak) prior information to select a solution. A different approach would be to assume that the unknown object f lies in a **subspace** of $\mathcal{L}_2$, *i.e.*, a linear combination of some basis functions. This type of assumption serves to bridge the continuous-discrete model (1.3.1) with the discrete-discrete model (1.2.2). Typically the basis functions are just equally spaced versions of a common kernel $\beta_0(x,y)$ as used in the following representation:

e,res,fbasis

$$f(x,y) = \sum_{m,n} f[m,n]\, \beta_0(x - m\triangle_{\mathrm{X}}, y - n\triangle_{\mathrm{Y}}), \tag{1.3.6}$$

where $\triangle_{\mathrm{X}}$ and $\triangle_{\mathrm{Y}}$ denote the spacing of the basis functions. Under such a model, the image restoration goal is to estimate the coefficients $\{f[m,n]\}$ from the measurements $\{g[m,n]\}$.

For example, a common assumption in signal processing is that f is a **band-limited** function. If so, then by the 2D **sampling theorem** [52, p. 238] there exist sample distances $\triangle_{\mathrm{X}}$ and $\triangle_{\mathrm{Y}}$ for which (1.3.6) is exact with an appropriate sinc kernel:

$$\begin{aligned}
\beta_0(x,y) &= \operatorname{sinc}_2(x/\triangle_{\mathrm{X}}, y/\triangle_{\mathrm{Y}}) \\
\operatorname{sinc}_2(x,y) &\triangleq \operatorname{sinc}(x)\operatorname{sinc}(y) \\
\operatorname{sinc}(x) &\triangleq \begin{cases} \frac{\sin(\pi x)}{\pi x}, & x \neq 0 \\ 1, & x = 0. \end{cases}
\end{aligned} \tag{1.3.7}$$

In general, however, (1.3.6) is merely an approximation.

Another common choice is to use 2D rectangular functions as basis functions: $\beta_0(x,y) = \operatorname{rect}_2(x/\triangle_{\mathrm{X}}, y/\triangle_{\mathrm{Y}})$, which correspond to rectangular (or square in the usual case when $\triangle_{\mathrm{X}} = \triangle_{\mathrm{Y}}$) pixels of uniform value $f[m,n]$.

Often the basis functions satisfy the **interpolation property**

$$\beta_0(m\triangle_{\mathrm{X}}, n\triangle_{\mathrm{Y}}) = \begin{cases} 1, & n = m = 0 \\ 0 & \forall m,n \in \mathbb{Z} - \{0,0\}, \end{cases}$$

in which case $f[m,n] = f(m\triangle_{\mathrm{X}}, n\triangle_{\mathrm{Y}})$. The sinc and rect examples both satisfy this property.

Substituting the object model (1.3.6) into the integral in (1.3.1) yields

$$\begin{aligned}
\bar{g}[m,n] &= \iint b(m,n;x,y)\, f(x,y) \,\mathrm{d}x \,\mathrm{d}y \\
&= \iint b(m,n;x,y) \left[\sum_{k,l} f[k,l]\, \beta_0(x - k\triangle_{\mathrm{X}}, y - l\triangle_{\mathrm{Y}}) \right] \mathrm{d}x \,\mathrm{d}y \\
&= \sum_{k,l} b[m,n;k,l]\, f[k,l],
\end{aligned} \tag{1.3.8}$$

where the “discrete-discrete” impulse response is

e,res,cd,b,mnkl

$$b[m,n;k,l] \triangleq \iint b(m,n;x,y)\, \beta_0(x - k\triangle_{\mathrm{X}}, y - l\triangle_{\mathrm{Y}})\, \mathrm{d}x\, \mathrm{d}y\,. \tag{1.3.9}$$

Suppose we further assume that the imaging system is shift invariant in the following sense:

e,res,band,shift

$$b(m,n;x,y) = b_0(x - m\triangle_{\mathrm{X}}, y - n\triangle_{\mathrm{Y}}), \tag{1.3.10}$$

for some 2D PSF $b_0(x,y)$. An imaging system having a system response that satisfies (1.3.10) has the property that shifting the input object by an integer number of sample distances causes a corresponding shift of the output image. The model (1.3.10) is often a reasonable approximation, with the caveat that (1.3.10) assumes implicitly that the detector sample spacing is matched to the object model. Under this assumption, the discrete-discrete impulse response is also shift invariant:

$$\begin{aligned} b[m,n;k,l] &= \iint b(m,n;x,y)\, \beta_0(x - k\triangle_{\mathrm{X}}, y - l\triangle_{\mathrm{Y}})\, \mathrm{d}x\, \mathrm{d}y \\ &= \iint b_0(x - m\triangle_{\mathrm{X}}, y - n\triangle_{\mathrm{Y}})\, \beta_0(x - k\triangle_{\mathrm{X}}, y - l\triangle_{\mathrm{Y}})\, \mathrm{d}x\, \mathrm{d}y \\ &= \iint b_0(x - (m-k)\triangle_{\mathrm{X}}, y - (m-l)\triangle_{\mathrm{Y}})\, \beta_0(x,y)\, \mathrm{d}x\, \mathrm{d}y \\ &= b[m-k, n-l; 0, 0], \end{aligned}$$

assuming appropriate limits for the integrals. With these assumptions, we have

e,res,cd,si

$$\bar{g}[m,n] = \sum_{k,l} b[m-k, n-l; 0, 0]\, f[k,l], \tag{1.3.11}$$

which is equivalent to (1.2.2) except for (nontrivial!) details about the summation limits, as detailed in §1.4. So we have established that there is at least one set of conditions (namely (1.3.6) and (1.3.10)) under which the continuous-discrete model (1.3.1) will (almost) simplify to the discrete-discrete model (1.2.2).

If the true $f(x,y)$ contains sharp edges, then the model (1.3.6) may lead to blurred estimates of those edges. To preserve such edges, it may be desirable to use a basis for $f(x,y)$ that has smaller pixels than the sample spacing of the measurements $g[m,n]$. This will result in an under-determined problem, but this can be addressed using edge-preserving regularization, as discussed in §1.10. Even if an image is band-limited, using the object model (1.3.6) with the sinc basis (1.3.7) is usually inconvenient for implementation. On the other hand, the expansion (1.3.5) also seems awkward because it is imaging system dependent. Chapter 10 describes various basis functions for representing a continuous object f in terms of a finite set of parameters or expansion coefficients.

s,res,mat1

1.4 Matrix-vector representations of convolution (s,res,mat1)

The convolution notation (1.2.3) is ubiquitous in signal processing, and corresponding Fourier concepts are commonly expressed in these terms. For example, the very important **convolution property** of the 2D **Fourier transform** can be written:

e,res,dsft2,conv

$$\bar{g}[m,n] = b[m,n] \ast\ast f[m,n] \overset{\mathrm{DSFT}}{\longleftrightarrow} \bar{G}(\Omega_1,\Omega_2) = B(\Omega_1,\Omega_2)\, F(\Omega_1,\Omega_2), \tag{1.4.1}$$

where the 2D (discrete space) **Fourier transform** (**DSFT**) of $b[m,n]$ is defined by:

e,res,dsft2

$$B(\Omega_1,\Omega_2) = \sum_{m,n} b[m,n]\, \mathrm{e}^{-\imath(\Omega_1 m + \Omega_2 n)}\,, \tag{1.4.2}$$

and Ω_1, Ω_2 denote **digital frequencies** having units radians/sample. However, the convolution notation becomes inconvenient when one wants to relate signal processing concepts to methods from numerical linear algebra or statistics. Convolution is a linear operator, so we can represent the convolution operator using **matrix-vector notation**. Matrix-vector notation is also more convenient for describing system models that generalize (1.2.3), such as shift variant imaging systems and tomographic imaging systems. Facility with both signal processing notation and matrix-vector representations is essential for working with image recovery problems, so this section describes in detail how to relate these representations.

1.4.1 1D matrix-vector representations

For simplicity we first consider a 1D version of the signal restoration problem. Consider a 1D convolution relationship for an input signal $f[n]$ as follows:

e,res,conv1

$$g[n] = \bar{g}[n] + \varepsilon[n], \quad \bar{g}[n] = b[n] \ast f[n] = \sum_k b[n-k]\, f[k], \quad n = 0, \dots, N-1. \tag{1.4.3}$$

We want to represent the preceding "DSP-style" formula in the following matrix-vector form

e,res,y=Ax+e

$$\boldsymbol{y} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{\varepsilon}, \tag{1.4.4}$$

where $\boldsymbol{y}$, $\boldsymbol{\varepsilon}$, and $\boldsymbol{x}$ are **column vectors** (with $\boldsymbol{y}$ and $\boldsymbol{x}$ possibly having somewhat different lengths), and $\boldsymbol{A}$ is a matrix whose entries depend solely on the values of $b[n]$. When stored in a computer, the vectors $\boldsymbol{y}$ and $\boldsymbol{x}$ must have finite length, whereas the convolution sum in (1.4.3) above can have arbitrary indices "on paper," even $\sum_{k=-\infty}^{\infty}$, but that is impractical for computation. There are various choices for $\boldsymbol{A}$ depending on how we handle the "**end conditions**," *i.e.*, depending on how we choose the limits of the summation in (1.4.3). In all cases we form the measurement vector $\boldsymbol{y}$ from the measurements $g[n]$ in the obvious way:

e,res,y=g

$$\boldsymbol{y} = \begin{bmatrix} y_1 \\ \vdots \\ y_N \end{bmatrix} = \begin{bmatrix} g[0] \\ \vdots \\ g[N-1] \end{bmatrix}, \tag{1.4.5}$$

i.e., $y_i = g[i-1]$. For consistency with matrix algebra conventions, the vector index i starts at 1, whereas for consistency with signal processing (and ANSI C) conventions, the sample index n starts at 0. We define the noise vector $\boldsymbol{\varepsilon}$ similarly.

For vectorizing the measurements, (1.4.5) is the only natural choice because N is a particular value depending on the measurement system. However, there is somewhat more flexibility when we choose $\boldsymbol{A}$ and $\boldsymbol{x}$, because these are merely models. For concreteness, we illustrate the various choices using a hypothetical 1D imaging system that has the following 3-point impulse response:

$$b[n] = b[-1]\,\delta[n+1] + b[0]\,\delta[n] + b[1]\,\delta[n-1],$$

where the **Kronecker impulse function** is denoted

e,kronecker

$$\delta[n] \triangleq \begin{cases} 1, & n = 0 \\ 0, & \text{otherwise.} \end{cases} \tag{1.4.6}$$

For each of the end conditions described below, Fig. 1.4.1 gives an example of the corresponding matrix $\boldsymbol{A}$.

1.4.1.1 Zero end conditions

The simplest model is to assume that $f[n]$ is zero for $n < 0$ and for $n \geq N$, and to use $\sum_{k=0}^{N-1}$ in the convolution sum (1.4.3). If this "zero end condition" (also known as a **Dirichlet boundary condition** [wiki]) assumption is appropriate, then natural choices for $\boldsymbol{x}$ and $\boldsymbol{A}$ are as follows:

e,res,A,zero

$$\boldsymbol{A} = \begin{bmatrix} b[0] & b[-1] & 0 & \cdots & 0 & 0 & 0 \\ b[1] & b[0] & b[-1] & 0 & \cdots & 0 & 0 \\ 0 & b[1] & b[0] & b[-1] & 0 & \cdots & 0 \\ & & & \ddots & & & \\ 0 & \cdots & 0 & b[1] & b[0] & b[-1] & 0 \\ 0 & 0 & \cdots & 0 & b[1] & b[0] & b[-1] \\ 0 & 0 & 0 & \cdots & 0 & b[1] & b[0] \end{bmatrix}, \qquad \boldsymbol{x} = \begin{bmatrix} f[0] \\ \vdots \\ f[N-1] \end{bmatrix}. \tag{1.4.7}$$

This $N \times N$ matrix $\boldsymbol{A}$ is **Toeplitz** [wiki] (constant along the diagonals), and there are reasonably fast algorithms for manipulating (*e.g.*, inverting) such matrices [53], [54]. In this 1D case, the elements $\{a_{ij}\}$ of $\boldsymbol{A}$ are related to a general impulse response function $b[n]$ as follows:

e,res,aij,bn

$$a_{ij} = b[i-j], \qquad i, j = 1, \ldots, N. \tag{1.4.8}$$

However, in the shift-invariant case, relatively rarely does one actually implement $\boldsymbol{A}$ literally as a matrix. Computing $\boldsymbol{A}\boldsymbol{x}$ yields the same result as convolving $b[n]$ and $f[n]$, so usually one would simply pass the elements of $b[n]$ and $f[n]$ to a convolution routine. So for shift-invariant problems, the matrix-vector representation is more for convenience of mathematical *analysis* than for *implementation*. On the other hand, for shift-variant problems, storing $\boldsymbol{A}$ as a matrix is often appropriate, although a **sparse matrix** representation [55, p. 78] is usually most efficient due to the small percentage of nonzero elements in $\boldsymbol{A}$ when $b[n]$ has just a few non-zero values.

s,res,mat1,extend

1.4.1.2 Extended end conditions

In many situations, *e.g.*, optical imaging, the measurements are influenced by a larger scene than the field of view of the aperture due to the spreading caused by the imaging system PSF. In such cases, assuming zero end conditions can be unrealistic and one may need to allow for objects of extended size (relative to the measurements) in the system model[4] [57]–[61]. With regularization (discussed later) or constraints, partial recovery of an extended object can be possible, *i.e.*, the object vector $\boldsymbol{x}$ can have somewhat more elements than the measurement vector $\boldsymbol{y}$. In such cases, natural choices for $\boldsymbol{x}$ and $\boldsymbol{A}$ include the following:

e,res,A,extend

$$\boldsymbol{A} = \begin{bmatrix} b[1] & b[0] & b[-1] & 0 & \cdots & 0 & 0 \\ 0 & b[1] & b[0] & b[-1] & 0 & \cdots & 0 \\ & & \ddots & \ddots & \ddots & & \\ 0 & 0 & \cdots & 0 & b[1] & b[0] & b[-1] \end{bmatrix}, \qquad \boldsymbol{x} = \begin{bmatrix} f[-1] \\ f[0] \\ \vdots \\ f[N-1] \\ f[N] \end{bmatrix}. \tag{1.4.9}$$

Here $\boldsymbol{A}$ is a $N \times (N + L - 1)$ rectangular matrix where L is the length of the impulse response ($L = 3$ for this particular $b[n]$).

In my view, these **extended end conditions** are the most realistic model in most restoration problems and should be used in practice whenever feasible. However, often engineers like to "cut corners" to save computation. The next sections describe the "replicated," "mirror" and "periodic" end conditions that are often used in practice, essentially as approximations to the extended end conditions.

Mat *If* `b` *is the row vector* $[b[-1]\;\; b[0]\;\; b[1]]$, *the* MATLAB *command* `A = convmtx(fliplr(b),N)` *generates this* $\boldsymbol{A}$. *Again, the matrix representation of convolution is more useful for analysis than for computation.*

Mat *This model is related to* MATLAB*'s* `valid` *option for the 2D convolution routine* `conv2`.

1.4.1.3 Replicated end conditions

If the zero end conditions are unrealistic but the extended end conditions are considered infeasible, an alternative is to assume that the signal beyond the field of view is similar to the signal at the borders of the field of view, *i.e.*, $f[n] = f[0]$ for $n < 0$ and $f[n] = f[N-1]$ for $n \geq N$ in 1D. For the case $L = 3$ this assumption corresponds to models where $f[-1] = f[0]$ and $f[N] = f[N-1]$. In matrix-vector form this leads to the same N-point $\boldsymbol{x}$ as in (1.4.7), but where $\boldsymbol{A}$ has the following $N \times N$ square form:

e,res,A,repeat

$$\boldsymbol{A} = \begin{bmatrix} b[0]+b[1] & b[-1] & 0 & \cdots & 0 & 0 \\ b[1] & b[0] & b[-1] & 0 & \cdots & 0 \\ & & \ddots & & & \\ 0 & \cdots & 0 & b[1] & b[0] & b[-1] \\ 0 & 0 & \cdots & 0 & b[1] & b[0]+b[-1] \end{bmatrix}. \tag{1.4.10}$$

All rows of this $\boldsymbol{A}$ have the same cumulative sum; this property is desirable in some cases.

1.4.1.4 Mirror end conditions

A variation on the previous approach is to use **mirror end conditions** [62, Box 2], also known as **reflective boundary conditions** or **Neumann boundary conditions** [63], where we assume that $f[-n] = f[n]$ and $f[N-1+n] = f[N-1-n]$ for $n = 0, \ldots, N-1$. Specifying the corresponding $\boldsymbol{A}$ is left to Problem 1.8.

1.4.1.5 Periodic end conditions

Another choice for end conditions in (1.4.3) is to assume that $f[n]$ is N-periodic, *i.e.*, that[5] $f[n] = f[n \bmod N]$. Periodic end conditions rarely (if ever) hold exactly in practical imaging systems, but are a very useful approximation for *analyzing* algorithms in the frequency-domain, as described in §1.4.3.1 below. Under this assumption, one can show (Problem 1.3) that the linear convolution (1.4.3) becomes a N-point **circular convolution**:

e,res,circl,cconv

$$\bar{g}[n] = \tilde{b}[n] \circledast_N f[n] \triangleq \sum_{k=0}^{N-1} \tilde{b}[(n-k) \bmod N]\, f[k], \qquad n = 0, \ldots, N-1, \tag{1.4.11}$$

where the periodic superposition of $b[n]$ is defined by

e,res,circl,bt

$$\tilde{b}[n] \triangleq \sum_l b[n - lN]\,. \tag{1.4.12}$$

Usually we assume that the blur kernel $b[n]$ is zero for $|n| \geq N$, in which case (1.4.12) simplifies to

$$\tilde{b}[n] = b[n] + b[n-N], \qquad n = 0, \ldots, N-1.$$

[4] This problem is so important in helical cone-beam tomography that it has been dubbed the "**long object problem**" [56].

[5] Here, the **modulo** operator is defined by $n \bmod N = n - N\lfloor n/N \rfloor$, where the **floor function** $\lfloor t \rfloor$ denotes the largest integer no greater than t. See §27.12.

$$\text{Zero: } \begin{bmatrix} 4 & 2 & 0 & 0 & 0 & 0 \\ 1 & 4 & 2 & 0 & 0 & 0 \\ 0 & 1 & 4 & 2 & 0 & 0 \\ 0 & 0 & 1 & 4 & 2 & 0 \\ 0 & 0 & 0 & 1 & 4 & 2 \\ 0 & 0 & 0 & 0 & 1 & 4 \end{bmatrix} \qquad \text{Extended: } \begin{bmatrix} 1 & 4 & 2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 4 & 2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 4 & 2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 4 & 2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 4 & 2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 4 & 2 \end{bmatrix}$$

$$\text{Replicated: } \begin{bmatrix} 5 & 2 & 0 & 0 & 0 & 0 \\ 1 & 4 & 2 & 0 & 0 & 0 \\ 0 & 1 & 4 & 2 & 0 & 0 \\ 0 & 0 & 1 & 4 & 2 & 0 \\ 0 & 0 & 0 & 1 & 4 & 2 \\ 0 & 0 & 0 & 0 & 1 & 6 \end{bmatrix} \qquad \text{Mirror: } \begin{bmatrix} 4 & 3 & 0 & 0 & 0 & 0 \\ 1 & 4 & 2 & 0 & 0 & 0 \\ 0 & 1 & 4 & 2 & 0 & 0 \\ 0 & 0 & 1 & 4 & 2 & 0 \\ 0 & 0 & 0 & 1 & 4 & 2 \\ 0 & 0 & 0 & 0 & 3 & 4 \end{bmatrix}$$

$$\text{Periodic/circulant: } \begin{bmatrix} 4 & 2 & 0 & 0 & 0 & 1 \\ 1 & 4 & 2 & 0 & 0 & 0 \\ 0 & 1 & 4 & 2 & 0 & 0 \\ 0 & 0 & 1 & 4 & 2 & 0 \\ 0 & 0 & 0 & 1 & 4 & 2 \\ 2 & 0 & 0 & 0 & 1 & 4 \end{bmatrix}$$

Figure 1.4.1: Concrete examples of system matrices $\boldsymbol{A}$ for 1D convolution with various end conditions, for $b[n] = \{2\ \underline{4}\ 1\} = 2\,\delta[n+1] + 4\,\delta[n] + \delta[n-1]$, for the case $N = 6$. fig,mat1

For the circular convolution model, the matrix-vector representation uses the same $\boldsymbol{x}$ as in (1.4.7), but the $N \times N$ system matrix $\boldsymbol{A}$ becomes

$$\boldsymbol{A} = \begin{bmatrix} \tilde{b}[0] & \tilde{b}[N-1] & \tilde{b}[N-2] & \cdots & \tilde{b}[2] & \tilde{b}[1] \\ \tilde{b}[1] & \tilde{b}[0] & \tilde{b}[N-1] & \tilde{b}[N-2] & \cdots & \tilde{b}[2] \\ & & \ddots & & & \\ \tilde{b}[N-1] & \tilde{b}[N-2] & \tilde{b}[N-3] & \cdots & \tilde{b}[1] & \tilde{b}[0] \end{bmatrix}. \tag{1.4.13}$$ e,res,circl,A

In this case $a_{ij} = \tilde{b}[(i-j) \bmod N]$. This matrix $\boldsymbol{A}$ is called **circulant**, and can be considered to be an approximation to the Toeplitz form (1.4.7).

See [64]–[66] for discussion of restoration using **anti-reflective boundary conditions**, and [67] for further alternatives.

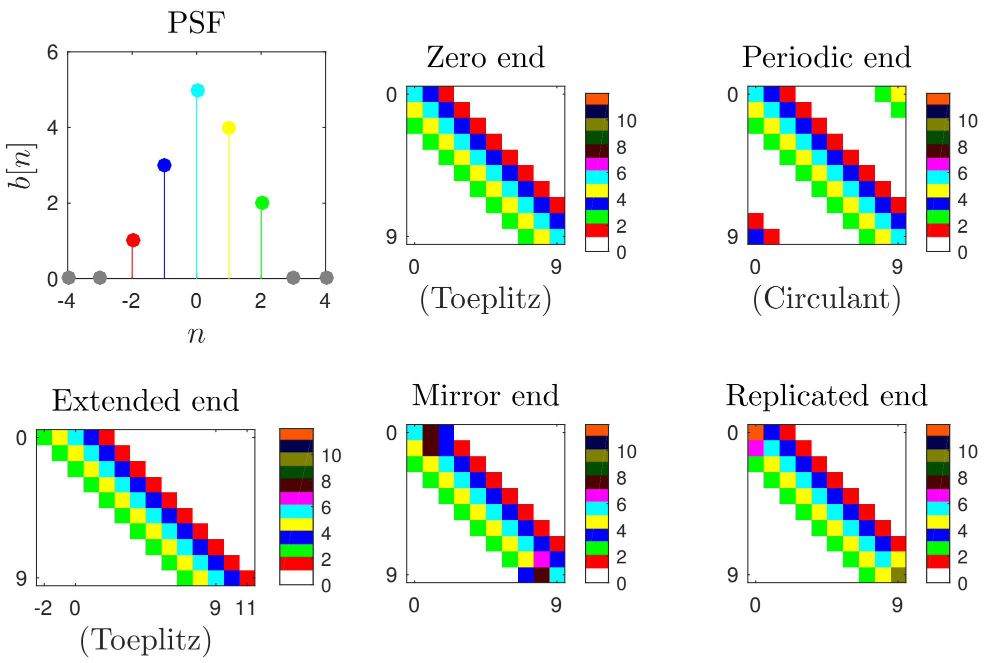


Figure 1.4.2: Illustration of system matrices $\boldsymbol{A}$ for 1D blur with different end conditions, for $N = 9$. The colors of the values used in $\boldsymbol{A}$ match those shown in the PSF $b[n]$. fig_res_mat1

s,res,mat2

1.4.2 2D matrix-vector representations (s,res,mat2)

For 2D (and higher dimensional) problems, the same general principles apply when representing convolutions with matrix-vector notation. The expressions relating the convolution form to the matrix-vector representation are messier, but, when expressed in matrix-vector form, both the 1D and 2D cases look like (1.4.4).

1.4.2.1 Lexicographic ordering

The first step is to represent a finite-sized image $g[m,n]$ using a "long" vector via **lexicographic ordering**. Specifically, if $g[m,n]$ has the $M \times N$ domain defined in (1.1.1), then the corresponding vector $\boldsymbol{y}$ is of length MN and the ith element of $\boldsymbol{y}$ is given by

e,res,lexico2

$$y_i = g[m(i), n(i)], \qquad i = 1, \ldots, MN, \tag{1.4.14}$$

where vector index i maps to pixel coordinates $[m(i), n(i)]$ as follows:

$$\begin{aligned} m(i) &\triangleq (i-1) \bmod M \\ n(i) &\triangleq \left\lfloor \frac{i-1}{M} \right\rfloor . \end{aligned} \tag{1.4.15}$$

Again the index i starts at 1, for consistency with matrix algebra, and the indices $[m,n]$ start at 0, for consistency with signal processing conventions. Spatial index $[m,n]$ corresponds to $i = 1 + m + nM$, *i.e.*,

e,res,mat2,i=1+m+nM

$$g[m,n] = y_i \Big|_{i=1+m+nM} . \tag{1.4.16}$$

The following figure summarizes the relationship between the vector $\boldsymbol{y}$ and the 2D image $g[m,n]$. Some authors define a **vec operator**, vec : $\mathbb{R}^{M \times N} \to \mathbb{R}^{MN}$, such that $\boldsymbol{y} = \text{vec}(g)$.

$y_1 = g[0,0]$	$y_2 = g[1,0]$	$\ldots$	$y_N = g[M-1,0]$
$y_{M+1} = g[0,1]$	$y_{M+2} = g[1,1]$	$\ldots$	$y_{2M} = g[M-1,1]$
		$\vdots$	
$y_{M(N-1)+1} = g[0,N-1]$	$y_{M(N-1)+2} = g[1,N-1]$	$\ldots$	$y_{MN} = g[M-1,N-1]$

$$\begin{tikzcd} \cdot \arrow[r] \arrow[d] & m \\ n \end{tikzcd}$$

Mat *MATLAB's* `reshape`, `colon`, `(:)`, `sub2ind`, *and* `ind2sub` *routines are useful for such conversions.*

We define vector $\boldsymbol{x}$ in terms of the **latent image** $f[m,n]$ in a manner similar to (1.4.14). The length of $\boldsymbol{x}$ depends on the boundary conditions used. For zero or periodic or mirror boundary conditions, $\boldsymbol{x}$ is a vector of length MN where $x_j = f[m(j), n(j)]$ where $[m(\cdot), n(\cdot)]$ were defined in (1.4.15). For extended end conditions the vector $\boldsymbol{x}$ is longer.

Using such lexicographic ordering, by linearity we can always write the 2D convolution expression (1.2.2), or the more general **superposition sum** (1.3.8), in the matrix-vector form (1.4.4), *i.e.*,

$$\boldsymbol{y} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{\varepsilon}.$$

The exact form of $\boldsymbol{A}$ depends again on the chosen end conditions.

1.4.2.2 Zero-end conditions

Consider the general discrete-space **superposition sum**

e,res,superposition2

$$\bar{g}[m,n] = \sum_{k,l} b[m,n;k,l]\, f[k,l], \qquad \begin{array}{l} m = 0, \ldots, M-1 \\ n = 0, \ldots, N-1. \end{array} \tag{1.4.17}$$

This sum characterizes the **input-output relationship** of any 2D discrete-space linear system, and this generality is needed for shift-variant systems. If one uses the end conditions that $f[m,n]$ is zero outside the domain m=0,…,M-1, n=0,…,N-1, then the corresponding system matrix $\boldsymbol{A}$ has size $MN \times MN$ and has entries

e,res,mat2,aij,sup

$$a_{ij} = b[m(i), n(i); m(j), n(j)], \tag{1.4.18}$$

where $m(\cdot)$ and $n(\cdot)$ were defined in (1.4.15). Expressed another way:

$$a_{1+m+nM,\, 1+m+nM} = b[m,n;k,l]$$

for $m, k = 0, \ldots, M-1$ and $n, l = 0, \ldots, N-1$. Any such matrix $\boldsymbol{A}$ has the following **block matrix** form:

e,res,mat2,block

$$\boldsymbol{A} = \begin{bmatrix} \boldsymbol{A}_{00} & \cdots & \boldsymbol{A}_{0,N-1} \\ \vdots & \ddots & \vdots \\ \boldsymbol{A}_{N-1,0} & \cdots & \boldsymbol{A}_{N-1,N-1} \end{bmatrix}, \tag{1.4.19}$$

where the $M \times M$ submatrix $\boldsymbol{A}_{nl}$ describes how the lth row of the input image contributes to the nth row of the output image and has elements

$$[\boldsymbol{A}_{nl}]_{mk} = b[m, n; k, l] \,.$$

See Fig. 1.4.3 for an example.

Other end conditions have similar expressions (see Problem 1.9) with different values for the elements of $\boldsymbol{A}$ corresponding to pixels at the borders of the image. In general, we can always write (1.4.17) as $\bar{\boldsymbol{y}} = \boldsymbol{A}\boldsymbol{x}$ using such an $\boldsymbol{A}$.

See [68] for statistical methods for dealing with boundary artifacts that arise from end condition choices.

Mat MATLAB's `convmtx2` *uses zero-end conditions but produces a sparse matrix that has more rows than columns.*

1.4.2.3 Shift-invariant 2D blur and Toeplitz matrices

If the 2D blur is shift invariant, *i.e.*, if $b[m, n; k, l] = b[m - k, n - l]$, then the superposition sum (1.4.17) becomes a convolution. In such cases, and for zero end conditions, the system matrix $\boldsymbol{A}$ has elements

e,res,mat2,aij,si

$$a_{ij} = b[m(i) - m(j), n(i) - n(j)], \tag{1.4.20}$$

and again has the block form (1.4.19) where the submatrices have elements

e,res,mat2,Anlmk

$$[\boldsymbol{A}_{nl}]_{mk} = b[m - k, n - l] \,. \tag{1.4.21}$$

Because of the $m - k$ dependence, each of the blocks in (1.4.19) is **Toeplitz** in this shift invariant case, so $\boldsymbol{A}$ is said to have **Toeplitz blocks**. Furthermore, because of the $n - l$ dependence, all of the blocks along each "diagonal" in the block form (1.4.19) are the same, so $\boldsymbol{A}$ is said to be **block Toeplitz**. Combined, we say any such $\boldsymbol{A}$ is **block Toeplitz with Toeplitz blocks** (**BTTB**) [54]. With a slight abuse of terminology, we will often simply call such matrices **Toeplitz**.

ASPIRE *The command* `wt gen` *with* `system 0` *generates a sparse* $\boldsymbol{A}$ *with this form.*

IRT *The object* `Glsi_blur` *represents such system matrices as operators.*

1.4.2.4 Separable 2D blur

If $b[m, n; k, l]$ is a **separable** PSF, *i.e.*, if $b[m, n; k, l] = b_1[m; k]\, b_2[n; l]$, then the superposition sum (1.4.17) can be grouped as follows:

$$\bar{g}[m, n] = \sum_l b_2[n; l] \left(\sum_k b_1[m; k]\, f[k, l] \right).$$

The inner summation operates on the first index of $f[m, n]$; the outer summation operates on the second index. In this case the elements of $\boldsymbol{A}$ have the separable form

$$a_{ij} = b_1[m(i); m(j)]\, b_2[n(i); n(j)]$$

and the resulting matrix $\boldsymbol{A}$ has the special structure

e,res,mat2,kron

$$\boldsymbol{A} = \boldsymbol{A}_2 \otimes \boldsymbol{A}_1, \tag{1.4.22}$$

where $\boldsymbol{A}_1 \in \mathbb{R}^{M \times M}$ is one of the 1D matrix representations in §1.4 for the 1D PSF $b_1[m; k]$, $\boldsymbol{A}_2 \in \mathbb{R}^{N \times N}$ is likewise defined in terms of $b_2[n; l]$, and "$\otimes$" denotes the **Kronecker product** defined in (26.1.12).

When $\boldsymbol{A}_1$ and $\boldsymbol{A}_2$ are each a Toeplitz matrix, such as when we use zero end conditions and a shift-invariant blur model, then $\boldsymbol{A}_2 \otimes \boldsymbol{A}_1$ is again **block Toeplitz with Toeplitz blocks**.

1.4.2.5 2D periodic end conditions

If we assume 2D periodic end conditions, *i.e.*, that $f[m, n] = f[m \bmod M, n \bmod N]$, also called **toroidal boundary conditions**, then the ordinary convolution (1.2.2) becomes a 2D **circular convolution**, defined by

e,res,mat2,Cconv

$$\bar{g}[m, n] = \tilde{b}[m, n] \circledast_{M,N} f[m, n] \triangleq \sum_{k=0}^{M-1} \sum_{l=0}^{N-1} \tilde{b}[(m - k) \bmod M, (n - l) \bmod N]\, f[k, l], \tag{1.4.23}$$

where

$$\tilde{b}[m, n] \triangleq \sum_k \sum_l b[m - kM, n - lN] \,.$$

Although periodic end conditions rarely (if ever) hold in practice, circulant models are convenient for *analysis*.

This circular convolution relationship has the matrix-vector representation $\bar{\boldsymbol{y}} = \boldsymbol{A}\boldsymbol{x}$ where the elements of the $MN \times MN$ matrix $\boldsymbol{A}$ are given by (*cf.* (1.4.20))

e,res,mat2,aij,circ2

$$a_{ij} = \tilde{b}[(m(i) - m(j)) \bmod M, (n(i) - n(j)) \bmod N], \qquad i, j = 1, \ldots, MN, \tag{1.4.24}$$

where $m(\cdot)$ and $n(\cdot)$ were defined in (1.4.15). Such a matrix is called **block circulant with circulant blocks** (**BCCB**). In particular, in this case $\boldsymbol{A}$ has the form (1.4.19) where

$$[\boldsymbol{A}_{nl}]_{mk} = \tilde{b}[(m - k) \bmod M, (n - l) \bmod N] \,.$$

x,res,mat2

Example 1.4.1 *See Fig. 1.4.3 for examples of the system matrix* $\boldsymbol{A}$ *for a small image with* $M = 6$ *and* $N = 8$*. Each black rectangle outlines one of the (typically* $N \times N$*) blocks* $\boldsymbol{A}_{nl} \in \mathbb{R}^{M \times M}$.

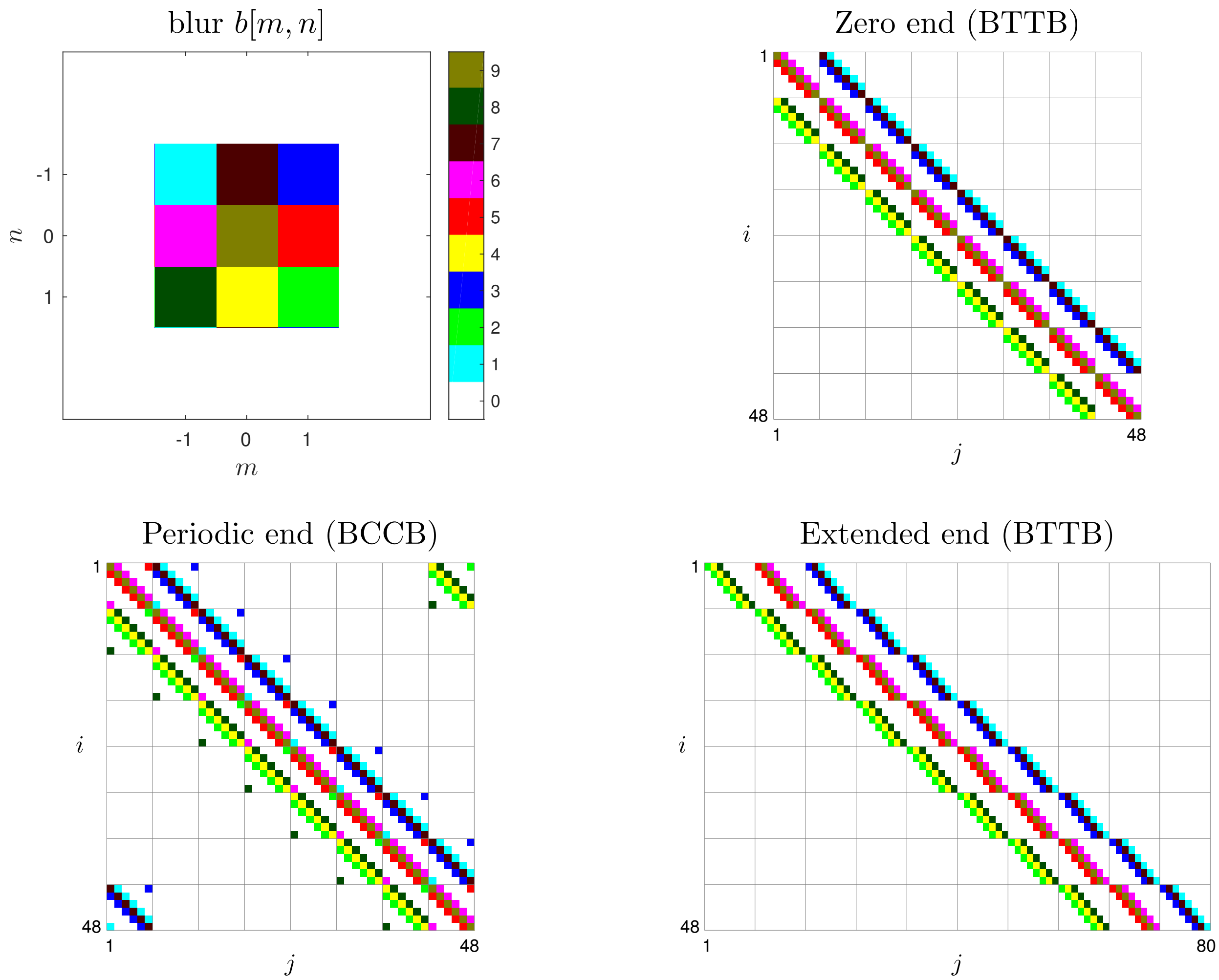


Figure 1.4.3: Illustration of system matrices $\boldsymbol{A}$ for 2D convolution for different end conditions, for $M = 6$ and $N = 8$. The upper-right figure illustrates (1.4.20), the lower-left figure illustrates (1.4.24), and the lower-right figure illustrates a 2D analogue of (1.4.9). Note that the spatial scale of the blur $b[m, n]$ figure differs from the others.

fig_res_mat2

[RQ]

s,res,circ

1.4.3 Circulant analysis of shift-invariant blur (s,res,circ)

For linear shift-invariant image restoration problems, $\boldsymbol{A}$ is always Toeplitz (or nearly Toeplitz depending on the chosen end conditions). However, most of the *methods* described in this *book* are also applicable to *shift-variant* problems, where the form of $\boldsymbol{A}$ depends on the physics and can be non-square and non-Toeplitz. For example, in tomography the elements of $\boldsymbol{A}$ correspond to a discretization of the **Radon transform**, which is usually non-square and hence non-Toeplitz. Despite the prevalence of non-Toeplitz problems, we can still obtain substantial insight by considering the Toeplitz case. In fact, we get the most signal processing *insight* by considering the case where $\boldsymbol{A}$ is circulant[6], *i.e.*, the special family of Toeplitz matrices that correspond to periodic end conditions. Analysis using circulant matrices helps us relate matrix algebra solutions to signal processing principles. Equally importantly, circulant approximations

[6] In analyses in later chapters, we will assume only that the square matrix $\boldsymbol{A}'\boldsymbol{A}$ is circulant, rather than making such an assumption about the possibly non-square matrix $\boldsymbol{A}$.

are useful for reducing computation by replacing large matrix operations with simpler **fast Fourier transform** (**FFT**) calculations.

The link between circulant matrices and circular convolution is the **convolution property** of the **discrete Fourier transform** (**DFT**). We show next that "**circular convolution**" and "**circulant matrix**" are extremely closely related concepts.

s,res,circ1

1.4.3.1 Circulant analysis in 1D (s,res,circ1)

In 1D, the **convolution property** of the **discrete Fourier transform** (**DFT**) can be expressed:

e,conv,dft

$$\bar{g}[n] = b[n] \circledast_N f[n] \overset{\text{DFT}}{\longleftrightarrow} \bar{G}_k = B_k F_k, \qquad (1.4.25)$$

where the N-point DFT of $b[n]$ is the following:

e,res,dft1

$$b[n] \overset{\text{DFT}}{\longleftrightarrow} B_k \triangleq \sum_{n=0}^{N-1} b[n] \, \mathrm{e}^{-\imath \frac{2\pi}{N} kn} = B(\Omega)\Big|_{\Omega = \frac{2\pi}{N} k}, \qquad k = 0, \ldots, N-1, \qquad (1.4.26)$$

where the corresponding **discrete-time Fourier transform** (**DTFT**) of $b[n]$ is

e,res,dtft

$$B(\Omega) \triangleq \sum_{n=-\infty}^{\infty} b[n] \, \mathrm{e}^{-\imath \Omega n} . \qquad (1.4.27)$$

Similarly F_k and $\bar{G}_k$ are defined in terms of $f[n]$ and $\bar{g}[n]$ respectively.

Mat *The convolution property (1.4.25) corresponds to the following* MATLAB *commands*

`g = ifft(fft(b) .* fft(f))`, or equivalently, `g = ifft(diag(fft(b)) * fft(f))`.

It is useful to rewrite (1.4.25) in matrix-vector notation as

$$\underbrace{\begin{bmatrix} \bar{G}_0 \\ \vdots \\ \bar{G}_{N-1} \end{bmatrix}}_{N \times 1} = \underbrace{\begin{bmatrix} B_0 & & \mathbf{0} \\ & \ddots & \\ \mathbf{0} & & B_{N-1} \end{bmatrix}}_{N \times N} \underbrace{\begin{bmatrix} F_0 \\ \vdots \\ F_{N-1} \end{bmatrix}}_{N \times 1} \text{ i.e., } \vec{\boldsymbol{G}} = \boldsymbol{\Gamma} \vec{\boldsymbol{F}},$$

where $\vec{\boldsymbol{G}} = (\bar{G}_0, \ldots, \bar{G}_{N-1})$ and $\vec{\boldsymbol{F}} = (F_0, \ldots, F_{N-1})$ are the vectors of N-point DFT coefficients of $\bar{g}[n]$ and $f[n]$ respectively, and $\boldsymbol{\Gamma} = \operatorname{diag}\{B_k\}$ is the $N \times N$ diagonal matrix with elements $B_0, \ldots, B_{N-1}$ along its diagonal, where the B_k values were defined in (1.4.26). Defining the vector $\bar{\boldsymbol{y}}$ in terms of $\bar{g}[n]$ as in (1.4.5) and $\boldsymbol{x}$ in terms of $f[n]$ as in (1.4.7), we have

$$\vec{\boldsymbol{G}} = \boldsymbol{Q} \, \bar{\boldsymbol{y}}, \quad \vec{\boldsymbol{F}} = \boldsymbol{Q} \boldsymbol{x},$$

where $\boldsymbol{Q}$ is the $N \times N$ **DFT matrix** having elements

e,res,qkn

$$q_{nk} = \mathrm{e}^{-\imath \frac{2\pi}{N} kn}, \qquad k, n = 0, \ldots, N-1. \qquad (1.4.28)$$

The inverse 1D N-point DFT corresponds to the inverse of this matrix and is given by

e,res,idft,1/N,fdft

$$\boldsymbol{Q}^{-1} = \frac{1}{N} \boldsymbol{Q}', \qquad (1.4.29)$$

where "$\boldsymbol{Q}'$" denotes the **Hermitian transpose** of $\boldsymbol{Q}$. Combining these relationships leads to a very useful tool for analysis, the following matrix-vector representation of the convolution property:

$$\bar{\boldsymbol{y}} = \boldsymbol{Q}^{-1} \boldsymbol{\Gamma} \boldsymbol{Q} \boldsymbol{x}.$$

Comparing the above relationships, we conclude that any circulant matrix, *e.g.*, the system matrix (1.4.13), has the following matrix decomposition:

e,res,A,dft

$$\boldsymbol{A} = \boldsymbol{Q}^{-1} \boldsymbol{\Gamma} \boldsymbol{Q} = \frac{1}{N} \boldsymbol{Q}' \boldsymbol{\Gamma} \boldsymbol{Q}. \qquad (1.4.30)$$

This is an eigenvector decomposition with eigenvalues $\{B_k\}$. So the eigenvalues of any circulant matrix are the DFT coefficients of the **first column** of that matrix. When viewing an expression like (1.4.30), one can think of $\boldsymbol{A}$ as a (circularly shift-invariant) filter whose frequency response is embedded in the diagonal elements of $\boldsymbol{\Gamma}$.

Mat *Although* $\boldsymbol{Q}$ *is used primarily for theoretical analysis, if needed it can be computed in* MATLAB *for modest values of* N *using* `Q = dftmtx(N)`.

s,res,circ2

1.4.3.2 Circulant analysis in 2D (s,res,circ2)

The frequency-domain expression corresponding to 2D **circular convolution** (1.4.23) is

$$G[k,l] = B[k,l]\, F[k,l], \qquad k = 0,\ldots,M-1,\ l = 0,\ldots,N-1,$$

where the $(M \times N)$-point **2D DFT** of $\tilde{b}[m,n]$ is defined as follows:

e,dft2

$$\begin{aligned} B[k,l] &= \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} \mathrm{e}^{-\imath(2\pi mk/M + 2\pi nl/N)}\, \tilde{b}[m,n] \\ &= \sum_{n=0}^{N-1} \mathrm{e}^{-\imath 2\pi nl/N} \left[\sum_{m=0}^{M-1} \mathrm{e}^{-\imath 2\pi mk/M}\, \tilde{b}[m,n] \right]. \end{aligned} \tag{1.4.31}$$

If we order lexicographically the 2D DFT coefficients $\{F[k,l]\}$ as a MN vector $\boldsymbol{F}$ and the 2D image $f[m,n]$ as a MN vector $\boldsymbol{x}$ per (1.4.14), then we can express the 2D DFT by a linear operator, *i.e.*, $\boldsymbol{F} = \boldsymbol{Q}\boldsymbol{x}$, where $\boldsymbol{Q}$ denotes the following $MN \times MN$ matrix

e,res,circ2,fdft

$$\boldsymbol{Q} = \boldsymbol{Q}_N \otimes \boldsymbol{Q}_M, \tag{1.4.32}$$

where $\boldsymbol{Q}_N$ denotes the 1D N-point DFT matrix defined in (1.4.28). The double sum in the 2D DFT expression (1.4.31) becomes the above **Kronecker product** when expressed in matrix notation. (See §26.1.6.) Thus, a circulant-block-circulant system matrix defined by (1.4.24) has the following eigenvector decomposition:

$$\boldsymbol{A} = \boldsymbol{Q}^{-1} \boldsymbol{\Gamma} \boldsymbol{Q} = \frac{1}{MN} \boldsymbol{Q}' \boldsymbol{\Gamma} \boldsymbol{Q},$$

where $\boldsymbol{\Gamma}$ is a diagonal matrix with diagonal elements that are the eigenvalues of $\boldsymbol{A}$, namely the 2D DFT coefficients $\{B[k,l]\}$ ordered lexicographically. So in 2D (and higher) we get the same form as (1.4.30) but with $\boldsymbol{Q}$ corresponding to a DFT of whatever dimension we are using. This decomposition is central to subsequent analyses.

Caution! In 1D the eigenvalues of a circulant matrix $\boldsymbol{A}$ were simply the 1D DFT coefficients of its first column. To find such eigenvalues in 2D, we take the first column of $\boldsymbol{A}$ and reshape it into a $M \times N$ array, and then compute the 2D DFT of that array. [RQ]

s,res,inv

1.5 Simple restoration methods (s,res,inv)

Having established two convenient notations, we return to the image restoration problem of recovering $f[m,n]$ from $g[m,n]$ under the model (1.2.3). We describe two "solutions," both of which are inadequate in general, for essentially the same reason, as will be analyzed.

1.5.1 The deconvolution solution

The **convolution property** of the Fourier transform (1.4.1), where $\bar{G}(\Omega_1,\Omega_2) = B(\Omega_1,\Omega_2)\, F(\Omega_1,\Omega_2)$, suggests the following **inverse-filter** solution:

e,res,1/H

$$\hat{F}(\Omega_1,\Omega_2) = \begin{cases} G(\Omega_1,\Omega_2) \,/\, B(\Omega_1,\Omega_2), & B(\Omega_1,\Omega_2) \neq 0 \\ ?, & B(\Omega_1,\Omega_2) = 0. \end{cases} \tag{1.5.1}$$

Equivalently,

$$\hat{f}[m,n] = b_{\mathrm{inv}}[m,n] \ast\ast\, g[m,n],$$

where $b_{\mathrm{inv}}[m,n]$ is the inverse Fourier transform of $1 / B(\Omega_1,\Omega_2)$. Such restoration methods are called **deconvolution**. Unfortunately, usually $b[m,n]$ is a lowpass type of filter, so $B(\Omega_1,\Omega_2)$ is zero or near zero for high spatial frequencies. So the simple inverse filter approach greatly amplifies high spatial-frequency noise. This property should be unsurprising because we did not consider noise when proposing (1.5.1).

Similar noise-amplifying phenomena are present in typical analytical solutions to tomographic reconstruction problems, because those solutions also ignore noise in the problem formulation. This book emphasizes methods where noise considerations are fundamental to the formulation.

1.5.2 The matrix inverse solution

By examining the matrix-vector expression (1.4.4), one could be tempted to propose the solution

e,res,x=A,inv,y

$$\hat{\boldsymbol{x}} = \boldsymbol{A}^{-1} \boldsymbol{y}, \tag{1.5.2}$$

at least when $\boldsymbol{A}$ is an invertible (and hence square) matrix [69]. We analyze this type of solution in more detail in Chapter 15, where the desirability of **regularization** is emphasized. For initial understanding of why this matrix inverse solution is inadequate, consider the case where $\boldsymbol{A}$ is circulant (*e.g.*, (1.4.13)) so that the decomposition (1.4.30) applies. Then we have $\boldsymbol{A}^{-1} = \boldsymbol{Q}^{-1}\boldsymbol{\Gamma}^{-1}\boldsymbol{Q}$ so

$$\hat{\boldsymbol{x}} = \boldsymbol{Q}^{-1}\boldsymbol{\Gamma}^{-1}\boldsymbol{Q}\boldsymbol{y}.$$

This solution is essentially the same as (1.5.1) because $\boldsymbol{Q}$ corresponds to the DFT and $\boldsymbol{\Gamma}^{-1}$ has reciprocals of samples of the system frequency response $B(\Omega_1, \Omega_2)$ along its diagonal. Solutions of the form (1.5.2) are known as **algebraic reconstruction techniques** because they are based on linear algebra concepts only. Hereafter we will focus on methods that use linear algebra combined with appropriate statistical considerations. [RQ] s,res,stat

1.6 Statistical image restoration (s,res,stat)

Statistical methods for image restoration improve on the simple methods described in §1.5 by incorporating both statistical models for the noise in the measurements $\boldsymbol{y}$ and prior knowledge about the unknown object $\boldsymbol{x}$.

If one has confidence in the statistical model for $\boldsymbol{y}$ (and perhaps in a statistical model for $\boldsymbol{x}$ as well), then a theoretically appealing approach is to apply the tools of estimation theory to find an estimator $\hat{\boldsymbol{x}} = \hat{\boldsymbol{x}}(\boldsymbol{y})$ of $\boldsymbol{x}$.

Even within the framework of statistical estimation one must choose between differing philosophical approaches. The most frequently studied methods are **maximum-likelihood** (ML) estimation, **Bayesian estimation**, and **penalized-likelihood** estimation, all of which are described next. These approaches apply both to image restoration and to image reconstruction.

s,res,noise

1.6.1 Noise models (s,res,noise)

This section summarizes a few of the statistical models for the measurement noise that are most popular in image restoration problems. Once again, the matrix-vector notation facilitates the presentation.

1.6.1.1 Additive gaussian noise

The most prevalent statistical model for image restoration problems corresponds to (1.2.3), *i.e.*,

e,res,y=A*x+e

$$\boldsymbol{y} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{\varepsilon}, \tag{1.6.1}$$

where[7] $\boldsymbol{y}, \boldsymbol{\varepsilon} \in \mathbb{R}^{n_{\mathrm{d}}}$, $\boldsymbol{x} \in \mathbb{R}^{n_{\mathrm{p}}}$, and $\boldsymbol{A} \in \mathbb{R}^{n_{\mathrm{d}} \times n_{\mathrm{p}}}$. For a $M \times N$ image, for most cases, including where zero end conditions are used, we have $n_{\mathrm{p}} = n_{\mathrm{d}} = MN$. But we may have $n_{\mathrm{d}} \neq n_{\mathrm{p}}$ for other end conditions, such as the extended end conditions, and for other image recovery problems.

To describe the noise statistics, we will often refer to the $n_{\mathrm{d}} \times n_{\mathrm{d}}$ **covariance matrix** of $\boldsymbol{\varepsilon}$, defined by

e,res,noise,cov

$$\boldsymbol{K}_{\boldsymbol{\varepsilon}} = \mathsf{Cov}\{\boldsymbol{\varepsilon}\} = \mathsf{E}[(\boldsymbol{\varepsilon} - \mathsf{E}[\boldsymbol{\varepsilon}])(\boldsymbol{\varepsilon} - \mathsf{E}[\boldsymbol{\varepsilon}])'], \tag{1.6.2}$$

where $\mathsf{E}[\cdot]$ denotes statistical expectation. The elements of $\boldsymbol{K}_{\boldsymbol{\varepsilon}}$ are given as follows:

e,res,noise,cov,ij

$$[\boldsymbol{K}_{\boldsymbol{\varepsilon}}]_{ij} = \boldsymbol{e}_i' \boldsymbol{K}_{\boldsymbol{\varepsilon}} \boldsymbol{e}_j = \mathsf{Cov}\{\varepsilon_i, \varepsilon_j\} = \mathsf{E}[(\varepsilon_i - \mathsf{E}[\varepsilon_i])(\varepsilon_j - \mathsf{E}[\varepsilon_j])], \tag{1.6.3}$$

where $\boldsymbol{e}_j$ denotes the jth **unit vector** of length n_{d}.

Usually one assumes that additive noise has a gaussian distribution with zero mean and a known covariance matrix $\boldsymbol{K}_{\boldsymbol{\varepsilon}}$ that is symmetric positive definite. Then (1.6.1) is equivalent to saying that $\boldsymbol{y}$ is a gaussian random vector with the following **probability density function** (**pdf**):

e,res,gauss,pdf

$$\mathsf{p}(\boldsymbol{y} \,|\, \boldsymbol{x}) = \frac{1}{\sqrt{(2\pi)^{n_{\mathrm{d}}} \det\{\boldsymbol{K}_{\boldsymbol{\varepsilon}}\}}} \exp\left(-\frac{1}{2}(\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x})' \boldsymbol{K}_{\boldsymbol{\varepsilon}}^{-1} (\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x})\right), \tag{1.6.4}$$

where $\det\{\boldsymbol{K}_{\boldsymbol{\varepsilon}}\}$ denotes the **determinant** of the noise **covariance matrix** $\boldsymbol{K}_{\boldsymbol{\varepsilon}}$.

1.6.1.2 Poisson measurements

For some illumination conditions, the random quantum effects of photons may dominate over other sources of measurement errors. This variability is called **shot noise** [wiki]. Often the statistics of these effects are modeled by Poisson distributions, so an alternative to (1.6.1) is to assume that the elements y_i of the measurement vector $\boldsymbol{y}$ are independent Poisson random variables whose means are given by the ith element of $\boldsymbol{A}\boldsymbol{x}$. To denote such independent Poisson variates, we write

e,res,poisson

$$y_i \sim \mathsf{Poisson}\{\bar{y}_i(\boldsymbol{x})\}, \quad i = 1, \ldots, n_{\mathrm{d}}, \tag{1.6.5}$$

where $y_i \in \{0, 1, 2, \ldots\}$ and where

$$\bar{y}_i(\boldsymbol{x}) \triangleq \mathsf{E}[y_i \,|\, \boldsymbol{x}] = [\boldsymbol{A}\boldsymbol{x}]_i = \sum_{j=1}^{n_{\mathrm{p}}} a_{ij} x_j.$$

Specifically, the shorthand (1.6.5) implies **statistical independence** and denotes the following assumption for the **probability mass function** (**PMF**) of $\boldsymbol{y}$:

e,res,noise,poisson

$$\mathsf{P}\{\boldsymbol{y} \,|\, \boldsymbol{x}\} = \prod_{i=1}^{n_{\mathrm{d}}} \mathsf{P}\{y_i \,|\, \boldsymbol{x}\} = \prod_{i=1}^{n_{\mathrm{d}}} \mathrm{e}^{-[\boldsymbol{A}\boldsymbol{x}]_i} ([\boldsymbol{A}\boldsymbol{x}]_i)^{y_i} / (y_i!), \tag{1.6.6}$$

where “!” denotes the factorial. This model is particularly appropriate for measurement systems that count photon interactions, as described in Chapter 8.

[7] The notation “$\boldsymbol{x} \in \mathbb{R}^{n_{\mathrm{p}}}$” indicates that $\boldsymbol{x}$ is a column vector of length n_{p}.

1.6.1.3 Poisson+gaussian measurements

The additive gaussian noise model of (1.6.1) is inexact in optical imaging systems such as those using **CCD arrays** [7], [8]. A more realistic model would account for the quantum effects of photons that interact with the detector, as well as the additional noise in the readout electronics. A combination of Poisson and gaussian distributions is appropriate for such systems, although the resulting distribution is somewhat inconvenient and requires approximation for practical implementation [7], [8].

Similar inconveniences arise in PET scans that are precorrected for random coincidences [70], [71], and in X-ray CT systems with current integrating detectors [72].

s,res,stat,ml

1.6.2 Maximum-likelihood estimation (s,res,stat,ml)

One can think of a statistical model $\mathsf{p}(\boldsymbol{y} \,|\, \boldsymbol{x})$ as quantifying the "agreement" between the measurement vector $\boldsymbol{y}$ and a candidate object vector $\boldsymbol{x}$. For **maximum-likelihood** estimation, one finds the $\boldsymbol{x}$ that maximizes this agreement, *i.e.*, the $\hat{\boldsymbol{x}}$ that best fits the data, using the **log-likelihood**

$$\mathsf{L}(\boldsymbol{x}) \triangleq \log \mathsf{p}(\boldsymbol{y} \,|\, \boldsymbol{x}) \,.$$

(Typically the dependence of L on the data $\boldsymbol{y}$ is suppressed notationally because we have only one vector $\boldsymbol{y}$ in a given experiment, but need to consider many candidate $\boldsymbol{x}$ vectors.) The ML estimator is defined by

$$\hat{\boldsymbol{x}}_{\mathrm{ML}} = \arg\max_{\boldsymbol{x}} \mathsf{L}(\boldsymbol{x}),$$

where the maximization is restricted to the set of acceptable values of $\boldsymbol{x}$. For example, often we only allow vectors $\boldsymbol{x}$ having nonnegative elements.

1.6.2.1 Poisson noise and the Richardson-Lucy iteration

The log-likelihood associated with the Poisson model (1.6.6) is

$$\mathsf{L}(\boldsymbol{x}) \equiv \sum_{i=1}^{n_{\mathrm{d}}} y_i \log([\boldsymbol{A}\boldsymbol{x}]_i) - [\boldsymbol{A}\boldsymbol{x}]_i = \boldsymbol{y}' \log(\boldsymbol{A}\boldsymbol{x}) - \boldsymbol{1}'\boldsymbol{A}\boldsymbol{x},$$

where we use the nonstandard (but convenient) notation "$\equiv$" to indicate that the two expressions are equal up to **irrelevant constants** that are independent of $\boldsymbol{x}$, and here log acts element-wise on a vector argument. Lucy considered this model [73], and derived the following iteration[8]:

e,res,stat,lr

$$x_j^{(n+1)} = \frac{x_j^{(n)}}{\sum_{i=1}^{n_{\mathrm{d}}} a_{ij}} \sum_{i=1}^{n_{\mathrm{d}}} a_{ij} y_i / [\boldsymbol{A}\boldsymbol{x}^{(n)}]_i. \qquad (1.6.7)$$

Richardson had earlier derived this same iteration using Bayes rule [74]. In the image restoration field, this is known as the **Richardson-Lucy** (or **Lucy-Richardson**) deconvolution procedure. It turns out to be the same formula as the ML-EM algorithm for emission tomography, as described in detail in Chapter 16. An appealing characteristic of the iteration (1.6.7) is that if the initial image $\boldsymbol{x}^{(0)}$ is nonnegative, then so are all the iterates $\{\boldsymbol{x}^{(n)}\}$. However, as the iterations proceed the images become increasingly noisy, due to the nature of unregularized ML restoration.

IRT *See* `eml_em.m`.

Mat *See* `deconvlucy`.

1.6.2.2 Gaussian noise

The limitations of the ML approach are illustrated easily by considering the gaussian model (1.6.4), for which the log-likelihood is

$$\begin{aligned}
\mathsf{L}(\boldsymbol{x}) &= -\frac{1}{2}(\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x})' \boldsymbol{K}_{\boldsymbol{\varepsilon}}^{-1} (\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}) - \frac{1}{2} \log(\det\{2\pi \boldsymbol{K}_{\boldsymbol{\varepsilon}}\}) \\
&\equiv -\frac{1}{2}(\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x})' \boldsymbol{K}_{\boldsymbol{\varepsilon}}^{-1} (\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}) = -\frac{1}{2} \left\| \boldsymbol{K}_{\boldsymbol{\varepsilon}}^{-1/2} (\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}) \right\|_2^2 .
\end{aligned}$$

This $\mathsf{L}(\boldsymbol{x})$ is a **concave function** [wiki] (see §27.9). If $\boldsymbol{A}$ has full column rank (which is rare in deblurring problems), then the ML estimate for gaussian noise is

e,res,stat,ml,wls

$$\hat{\boldsymbol{x}}_{\mathrm{ML}} = \left[\boldsymbol{A}' \boldsymbol{K}_{\boldsymbol{\varepsilon}}^{-1} \boldsymbol{A}\right]^{-1} \boldsymbol{A}' \boldsymbol{K}_{\boldsymbol{\varepsilon}}^{-1} \boldsymbol{y}. \qquad (1.6.8)$$

[8] Lucy's derivation assumed $\sum_{i=1}^{n_{\mathrm{d}}} a_{ij} = 1$.

In addition to having full column rank, if $\boldsymbol{A}$ is also square then it is invertible [wiki], and the unconstrained maximizer of L is

e,res,xh,ml,inv

$$\hat{\boldsymbol{x}}_{\mathrm{ML}} = \boldsymbol{A}^{-1} \boldsymbol{y}. \tag{1.6.9}$$

So the ML estimation criterion can again lead to the "inverse" solution (1.5.2), yielding unacceptably noisy images in most practical problems.

To reduce this noise, one must impose some type of **constraints** on the estimator $\hat{\boldsymbol{x}}$, or otherwise incorporate prior information about $\boldsymbol{x}$. Bayesian estimation and penalized-likelihood estimation are two related ways to achieve this goal. [RQ]

s,res,stat,bayes

1.7 Bayesian estimation (s,res,stat,bayes)

For ML estimation, the only statistical model required is that of the measurements, namely the likelihood $\mathsf{p}(\boldsymbol{y} \,|\, \boldsymbol{x})$. For **Bayesian estimation**, one must also postulate a probability distribution $\mathsf{p}(\boldsymbol{x})$ for the unknown object vectors. This distribution is called the **prior distribution** for $\boldsymbol{x}$, because it describes object properties that are assumed to be "known" *before* the measurements $\boldsymbol{y}$ are acquired.

1.7.1 MMSE estimation

Given a statistical model $\mathsf{p}(\boldsymbol{y} \,|\, \boldsymbol{x})$ and a prior $\mathsf{p}(\boldsymbol{x})$, in the Bayesian framework one can devise estimators that minimize an expected cost, called the **risk**, averaged over the family of possible objects $\boldsymbol{x}$. The simplest risk function is simply the **mean-squared error** (**MSE**) of an estimator $\hat{\boldsymbol{x}}$, defined by

$$\mathrm{MSE}(\hat{\boldsymbol{x}}) = \mathsf{E}\left[\|\hat{\boldsymbol{x}} - \boldsymbol{x}\|^2\right] = \iint \|\hat{\boldsymbol{x}}(\boldsymbol{y}) - \boldsymbol{x}\|^2 \, \mathsf{p}(\boldsymbol{y} \,|\, \boldsymbol{x}) \, \mathsf{p}(\boldsymbol{x}) \, \mathrm{d}\boldsymbol{y} \, \mathrm{d}\boldsymbol{x} \,.$$

A classical result in Bayesian estimation theory [75, p. 419] is that the **minimum mean-squared error** (**MMSE**) estimator for $\boldsymbol{x}$ given $\boldsymbol{y}$ is the following **conditional expectation** or **conditional mean**:

e,res,mmse

$$\hat{\boldsymbol{x}}_{\mathrm{MMSE}} = \arg\min_{\hat{\boldsymbol{x}}} \mathrm{MSE}(\hat{\boldsymbol{x}}) = \mathsf{E}[\boldsymbol{x} \,|\, \boldsymbol{y}] = \int \boldsymbol{x} \, \mathsf{p}(\boldsymbol{x} \,|\, \boldsymbol{y}) \, \mathrm{d}\boldsymbol{x} \,. \tag{1.7.1}$$

Unfortunately this conditional expectation is difficult to compute for many problems, so MMSE estimation is used only in relatively rare cases.

In general the MMSE estimator is nonlinear. If this is undesirable or intractable, another option is to try to find the *linear* or *affine* estimator that minimizes MMSE. This too can be challenging in cases where the MSE is itself analytically intractable.

1.7.2 MAP estimation

In part because of these difficulties, the Bayesian estimator that is often used in practice is the **maximum *a posteriori*** (**MAP**) approach, defined as the maximizer of the **posterior distribution** $\mathsf{p}(\boldsymbol{x} \,|\, \boldsymbol{y})$ as follows:

$$\hat{\boldsymbol{x}}_{\mathrm{MAP}} = \arg\max_{\boldsymbol{x}} \mathsf{p}(\boldsymbol{x} \,|\, \boldsymbol{y}) \,.$$

This approach finds the image that has the highest posterior probability[9] given the data $\boldsymbol{y}$. By Bayes' rule:

$$\mathsf{p}(\boldsymbol{x} \,|\, \boldsymbol{y}) = \frac{\mathsf{p}(\boldsymbol{y} \,|\, \boldsymbol{x}) \, \mathsf{p}(\boldsymbol{x})}{\mathsf{p}(\boldsymbol{y})},$$

where, by **total probability** [wiki], the overall data distribution $\mathsf{p}(\boldsymbol{y}) = \int \mathsf{p}(\boldsymbol{y} \,|\, \tilde{\boldsymbol{x}}) \, \mathsf{p}(\tilde{\boldsymbol{x}}) \, \mathrm{d}\tilde{\boldsymbol{x}}$ is independent of the true, unknown value of the image $\boldsymbol{x}$. Because the logarithm function is monotone, an equivalent formulation is:

e,res,map

$$\hat{\boldsymbol{x}}_{\mathrm{MAP}} = \arg\max_{\boldsymbol{x}} \left[\log \mathsf{p}(\boldsymbol{y} \,|\, \boldsymbol{x}) + \log \mathsf{p}(\boldsymbol{x})\right], \tag{1.7.2}$$

where we can ignore $\log \mathsf{p}(\boldsymbol{y})$ because it is independent of $\boldsymbol{x}$.

In a few special cases there are analytical expressions for the MAP estimator. However, most interesting problems lack such an analytical solution, so an iterative algorithm is needed to perform the maximization. Such iterative algorithms are a primary focus of this *book*.

A philosophical difficulty with the Bayesian paradigm is that the "priors" $\mathsf{p}(\boldsymbol{x})$ that are commonly used in Bayesian image restoration methods capture only very local properties of images. If one generates random draws from typical Bayesian image priors, the results look very little like natural images. (See Fig. 1.7.1.) This is contrary to the usual sense of the term "prior" in statistical estimation, where values drawn from the prior distribution are representative of the objects under investigation.

[9] The MAP estimator also minimizes the "**hit-or-miss**" risk function [76, p. 343]. The suitability of that risk function for imaging problems is somewhat debatable because it assigns the same loss to *all* incorrect estimates.

s,res,stat,gauss

1.7.3 Bayesian estimation in linear gaussian models (s,res,stat,gauss)

As a concrete example of Bayesian estimation, we consider the linear model (1.6.1) with additive gaussian noise and assume that the prior distribution for $\boldsymbol{x}$ is also gaussian. This example is one of the few where analytical solutions are available. The MAP and MMSE estimators turn out to be identical in this problem, unlike in more general cases.

Specifically, consider the following statistical assumptions.

- $\boldsymbol{y} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{\varepsilon}$ with $\boldsymbol{A} \in \mathbb{R}^{n_\mathrm{d} \times n_\mathrm{p}}$
- $\boldsymbol{\varepsilon} \sim \mathsf{N}(\boldsymbol{0}, \boldsymbol{K}_{\boldsymbol{\varepsilon}})$
- $\boldsymbol{x} \sim \mathsf{N}(\boldsymbol{\mu}_{\boldsymbol{x}}, \boldsymbol{K}_{\boldsymbol{x}})$
- $\boldsymbol{\varepsilon}$ and $\boldsymbol{x}$ are independent
- $\boldsymbol{\mu}_{\boldsymbol{x}}$, $\boldsymbol{K}_{\boldsymbol{\varepsilon}}$, and $\boldsymbol{K}_{\boldsymbol{x}}$ are all known.

Due to the independence of $\boldsymbol{x}$ and $\boldsymbol{\varepsilon}$, the likelihood is:

$$\begin{aligned}
\mathsf{p}(\boldsymbol{y} \,|\, \boldsymbol{x}) &= \mathsf{p}_{\boldsymbol{y}|\boldsymbol{x}}(\boldsymbol{A}\boldsymbol{x} + \boldsymbol{\varepsilon} \,|\, \boldsymbol{x}) = \mathsf{p}_{\boldsymbol{\varepsilon}}(\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}) \\
&= \frac{1}{\sqrt{(2\pi)^{n_\mathrm{d}} \det\{\boldsymbol{K}_{\boldsymbol{\varepsilon}}\}}} \mathrm{e}^{-\frac{1}{2}(\boldsymbol{y}-\boldsymbol{A}\boldsymbol{x})' \boldsymbol{K}_{\boldsymbol{\varepsilon}}^{-1} (\boldsymbol{y}-\boldsymbol{A}\boldsymbol{x})}. \qquad (1.7.3)
\end{aligned}$$

By assumption, the prior distribution for $\boldsymbol{x}$ is:

e,res,px,gauss

$$\mathsf{p}(\boldsymbol{x}) = \frac{1}{\sqrt{(2\pi)^{n_\mathrm{p}} \det\{\boldsymbol{K}_{\boldsymbol{x}}\}}} \mathrm{e}^{-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu}_{\boldsymbol{x}})' \boldsymbol{K}_{\boldsymbol{x}}^{-1} (\boldsymbol{x}-\boldsymbol{\mu}_{\boldsymbol{x}})}. \qquad (1.7.4)$$

s,res,map,gauss

1.7.3.1 MAP estimator

Combining the likelihood and the prior leads to the following form for the MAP estimator:

$$\begin{aligned}
\hat{\boldsymbol{x}}_{\mathrm{MAP}} &= \arg\max_{\boldsymbol{x}} \left[\log \mathsf{p}(\boldsymbol{y} \,|\, \boldsymbol{x}) + \log \mathsf{p}(\boldsymbol{x})\right] \\
&= \arg\min_{\boldsymbol{x}} \Psi(\boldsymbol{x}),
\end{aligned}$$

where we ignore irrelevant constants in defining the following **cost function** (also known as a **loss function** [wiki]):

e,res,Kx,map,gauss

$$\Psi(\boldsymbol{x}) \triangleq \frac{1}{2}(\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x})' \boldsymbol{K}_{\boldsymbol{\varepsilon}}^{-1} (\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}) + \frac{1}{2}(\boldsymbol{x} - \boldsymbol{\mu}_{\boldsymbol{x}})' \boldsymbol{K}_{\boldsymbol{x}}^{-1} (\boldsymbol{x} - \boldsymbol{\mu}_{\boldsymbol{x}}). \qquad (1.7.5)$$

This cost function consists of a data fit term and a prior term. By expanding $\Psi(\boldsymbol{x})$ and "**completing the square**," one can show that

$$\Psi(\boldsymbol{x}) \equiv \frac{1}{2} \left\| \boldsymbol{H}^{1/2}\boldsymbol{x} - \boldsymbol{H}^{-1/2}(\boldsymbol{A}'\boldsymbol{K}_{\boldsymbol{\varepsilon}}^{-1}\boldsymbol{y} + \boldsymbol{K}_{\boldsymbol{x}}^{-1}\boldsymbol{\mu}_{\boldsymbol{x}}) \right\|^2,$$

where[10] $\boldsymbol{H} \triangleq \boldsymbol{A}'\boldsymbol{K}_{\boldsymbol{\varepsilon}}^{-1}\boldsymbol{A} + \boldsymbol{K}_{\boldsymbol{x}}^{-1}$. It is clear from the preceding expression that a minimizer of $\Psi(\boldsymbol{x})$ is

$$\begin{aligned}
\hat{\boldsymbol{x}}_{\mathrm{MAP}} &= \boldsymbol{H}^{-1}(\boldsymbol{A}'\boldsymbol{K}_{\boldsymbol{\varepsilon}}^{-1}\boldsymbol{y} + \boldsymbol{K}_{\boldsymbol{x}}^{-1}\boldsymbol{\mu}_{\boldsymbol{x}}) \\
&= \left[\boldsymbol{A}'\boldsymbol{K}_{\boldsymbol{\varepsilon}}^{-1}\boldsymbol{A} + \boldsymbol{K}_{\boldsymbol{x}}^{-1}\right]^{-1} (\boldsymbol{A}'\boldsymbol{K}_{\boldsymbol{\varepsilon}}^{-1}\boldsymbol{y} + \boldsymbol{K}_{\boldsymbol{x}}^{-1}\boldsymbol{\mu}_{\boldsymbol{x}}) \\
&= \boldsymbol{\mu}_{\boldsymbol{x}} + \left[\boldsymbol{A}'\boldsymbol{K}_{\boldsymbol{\varepsilon}}^{-1}\boldsymbol{A} + \boldsymbol{K}_{\boldsymbol{x}}^{-1}\right]^{-1} \boldsymbol{A}'\boldsymbol{K}_{\boldsymbol{\varepsilon}}^{-1}(\boldsymbol{y} - \boldsymbol{A}\boldsymbol{\mu}_{\boldsymbol{x}}). \qquad (1.7.6)
\end{aligned}$$

Finding the minimizer of a function by "completing the square" is rarely a convenient approach. An alternative is to equate the gradient of $\Psi(\boldsymbol{x})$ to zero, provided no constraints such as nonnegativity are desired. Let ∇f denote the **row gradient** of a function $f : \mathbb{R}^{n_\mathrm{p}} \to \mathbb{R}$, *i.e.*,

$$\nabla f(\boldsymbol{x}) = \left[\frac{\partial}{\partial x_1} f(\boldsymbol{x}) \;\ldots\; \frac{\partial}{\partial x_{n_\mathrm{p}}} f(\boldsymbol{x}) \right].$$

Then one can easily verify the following very useful gradient relationships

- $\nabla_{\boldsymbol{x}}\, \boldsymbol{v}'\,\boldsymbol{x} = \boldsymbol{v}'$ for $\boldsymbol{v} \in \mathbb{R}^{n_\mathrm{p}}$
- $\nabla_{\boldsymbol{x}} \frac{1}{2}\boldsymbol{x}'\boldsymbol{M}\boldsymbol{x} = \boldsymbol{x}'\frac{1}{2}(\boldsymbol{M} + \boldsymbol{M}')$ for $\boldsymbol{M} \in \mathbb{R}^{n_\mathrm{p} \times n_\mathrm{p}}$.

We let $\bar{\nabla}$ denote the **column gradient** operator, the transpose of ∇. It follows that

$$\begin{aligned}
&\bar{\nabla}_{\boldsymbol{x}} \frac{1}{2}(\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x})'\boldsymbol{K}_{\boldsymbol{\varepsilon}}^{-1}(\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}) \\
&\quad = \bar{\nabla}_{\boldsymbol{x}} \left[\frac{1}{2}\boldsymbol{y}'\boldsymbol{K}_{\boldsymbol{\varepsilon}}^{-1}\boldsymbol{y} - \boldsymbol{y}'\boldsymbol{K}_{\boldsymbol{\varepsilon}}^{-1}\boldsymbol{A}\boldsymbol{x} + \frac{1}{2}\boldsymbol{x}'\boldsymbol{A}'\boldsymbol{K}_{\boldsymbol{\varepsilon}}^{-1}\boldsymbol{A}\boldsymbol{x} \right] \\
&\quad = -\boldsymbol{A}'\boldsymbol{K}_{\boldsymbol{\varepsilon}}^{-1}\boldsymbol{y} + \boldsymbol{A}'\boldsymbol{K}_{\boldsymbol{\varepsilon}}^{-1}\boldsymbol{A}\boldsymbol{x}
\end{aligned}$$

[10] The matrix $\boldsymbol{H}$ is invertible because $\boldsymbol{A}'\boldsymbol{K}_{\boldsymbol{\varepsilon}}^{-1}\boldsymbol{A}$ is positive semidefinite and $\boldsymbol{K}_{\boldsymbol{x}}^{-1}$ is positive definite so their sum is positive definite.

$$= \quad -\boldsymbol{A}'\boldsymbol{K}_{\boldsymbol{\varepsilon}}^{-1}(\boldsymbol{y}-\boldsymbol{A}\boldsymbol{x}). \tag{1.7.7}$$

Using these properties, the column gradient of $\Psi(\boldsymbol{x})$ defined in (1.7.5) above is

$$\begin{aligned}\nabla\,\Psi(\boldsymbol{x}) &= -\boldsymbol{A}'\boldsymbol{K}_{\boldsymbol{\varepsilon}}^{-1}(\boldsymbol{y}-\boldsymbol{A}\boldsymbol{x})+\boldsymbol{K}_{\boldsymbol{x}}^{-1}(\boldsymbol{x}-\boldsymbol{\mu}_{\boldsymbol{x}}) \\ &= -\boldsymbol{A}'\boldsymbol{K}_{\boldsymbol{\varepsilon}}^{-1}(\boldsymbol{y}-\boldsymbol{A}\boldsymbol{\mu}_{\boldsymbol{x}})+[\boldsymbol{A}'\boldsymbol{K}_{\boldsymbol{\varepsilon}}^{-1}\boldsymbol{A}+\boldsymbol{K}_{\boldsymbol{x}}^{-1}](\boldsymbol{x}-\boldsymbol{\mu}_{\boldsymbol{x}}).\end{aligned}$$

Equating this gradient to zero yields the MAP estimator

$$\hat{\boldsymbol{x}}_{\text{MAP}} = \boldsymbol{\mu}_{\boldsymbol{x}} + \left[\boldsymbol{A}'\boldsymbol{K}_{\boldsymbol{\varepsilon}}^{-1}\boldsymbol{A}+\boldsymbol{K}_{\boldsymbol{x}}^{-1}\right]^{-1}\boldsymbol{A}'\boldsymbol{K}_{\boldsymbol{\varepsilon}}^{-1}(\boldsymbol{y}-\boldsymbol{A}\boldsymbol{\mu}_{\boldsymbol{x}}),$$

which is equivalent to (1.7.6).

What happens to the MAP estimator as our confidence in the prior information decreases? If $\boldsymbol{K}_{\boldsymbol{x}} \to \infty$, then $\boldsymbol{K}_{\boldsymbol{x}}^{-1} \to \boldsymbol{0}$ and

$$\hat{\boldsymbol{x}}_{\text{MAP}} \to \left[\boldsymbol{A}'\boldsymbol{K}_{\boldsymbol{\varepsilon}}^{-1}\boldsymbol{A}\right]^{-1}\boldsymbol{A}'\boldsymbol{K}_{\boldsymbol{\varepsilon}}^{-1}\boldsymbol{y},$$

which is the conventional **weighted least-squares** (**WLS**) estimator, which also happens to be the ML estimator in this case. In particular, if $\boldsymbol{K}_{\boldsymbol{\varepsilon}} = \sigma^2\boldsymbol{I}$ and $\boldsymbol{A}$ is invertible, then $\hat{\boldsymbol{x}}_{\text{MAP}} \to \boldsymbol{A}^{-1}\boldsymbol{y}$, the inverse estimator. So the prior information is essential to avoid the noise problems associated with ML estimation.

A significant practical problem with this analytical solution is the apparent need to "invert" large matrices. In practice we usually minimize $\Psi(\boldsymbol{x})$ with an iterative algorithm, *e.g.*, Chapter 14, rather than using the explicit analytical form.

1.7.3.2 MMSE estimator

Because $\boldsymbol{x}$ and $\boldsymbol{y}$ are jointly gaussian under the assumptions in this example, it is straightforward to evaluate the conditional mean estimator (1.7.1). In general for jointly gaussian random vectors [77, p. 302] [76, p. 325], the conditional mean has the following form:

$$\mathsf{E}[\boldsymbol{x} \mid \boldsymbol{y}] = \mathsf{E}[\boldsymbol{x}] + \mathsf{Cov}\{\boldsymbol{x},\boldsymbol{y}\}\,\mathsf{Cov}\{\boldsymbol{y}\}^{-1}(\boldsymbol{y}-\mathsf{E}[\boldsymbol{y}]).$$

In this case, $\mathsf{Cov}\{\boldsymbol{x},\boldsymbol{y}\} = \mathsf{Cov}\{\boldsymbol{x},\boldsymbol{A}\boldsymbol{x}+\boldsymbol{\varepsilon}\} = \boldsymbol{K}_{\boldsymbol{x}}\boldsymbol{A}'$ and $\mathsf{Cov}\{\boldsymbol{y}\} = \mathsf{Cov}\{\boldsymbol{A}\boldsymbol{x}+\boldsymbol{\varepsilon}\} = \boldsymbol{A}\boldsymbol{K}_{\boldsymbol{x}}\boldsymbol{A}'+\boldsymbol{K}_{\boldsymbol{\varepsilon}}$, so the MMSE estimator has the following affine form:

e,res,stat,gauss,mmse

$$\hat{\boldsymbol{x}}_{\text{MMSE}} = \boldsymbol{\mu}_{\boldsymbol{x}} + \boldsymbol{K}_{\boldsymbol{x}}\boldsymbol{A}'\left[\boldsymbol{A}\boldsymbol{K}_{\boldsymbol{x}}\boldsymbol{A}'+\boldsymbol{K}_{\boldsymbol{\varepsilon}}\right]^{-1}(\boldsymbol{y}-\boldsymbol{A}\boldsymbol{\mu}_{\boldsymbol{x}}). \tag{1.7.8}$$

Using the **matrix inversion lemma** (see (26.1.9) and (26.1.10)), one can show that this expression is identical to (1.7.6). Thus, the MAP and MMSE estimators are identical in this linear gaussian problem. This gives further motivation for using MAP estimation because many problems are "nearly" gaussian.

res,stat,gauss,xmean=0

1.7.3.3 Interpretation of MAP/MMSE estimators

For the linear gaussian model with a gaussian prior, we can rewrite the MAP/MMSE estimators in (1.7.6) and (1.7.8) as follows:

$$\hat{\boldsymbol{x}}_{\text{MAP/MMSE}} = \left[\boldsymbol{A}'\boldsymbol{K}_{\boldsymbol{\varepsilon}}^{-1}\boldsymbol{A}+\boldsymbol{K}_{\boldsymbol{x}}^{-1}\right]^{-1}\boldsymbol{A}'\boldsymbol{K}_{\boldsymbol{\varepsilon}}^{-1}\boldsymbol{y} + \left(\boldsymbol{I}-\left[\boldsymbol{A}'\boldsymbol{K}_{\boldsymbol{\varepsilon}}^{-1}\boldsymbol{A}+\boldsymbol{K}_{\boldsymbol{x}}^{-1}\right]^{-1}\boldsymbol{A}'\boldsymbol{K}_{\boldsymbol{\varepsilon}}^{-1}\boldsymbol{A}\right)\boldsymbol{\mu}_{\boldsymbol{x}}.$$

The first term depends on the data, whereas the second term depends on the prior mean image $\boldsymbol{\mu}_{\boldsymbol{x}}$. In other words, the second term is unrelated to the measurements, so it could be viewed before taking any data! That term would be important quantitatively in terms of minimizing MSE, but qualitatively it provides no useful "new" information. All of the useful information about the scene (or patient in the medical context) is in the first term that depends on the data $\boldsymbol{y}$. Therefore in my view it is best to simply choose $\boldsymbol{\mu}_{\boldsymbol{x}} = \boldsymbol{0}$ (or adopt another approach altogether) in which case the second term disappears. Indeed most modern image reconstruction methods do not include any $\boldsymbol{\mu}_{\boldsymbol{x}}$ term. The first term above also depends on the prior covariance $\boldsymbol{K}_{\boldsymbol{x}}$, and that prior can strongly influence the image quality as shown below.

1.7.3.4 White gaussian case

The simplest special case of this MAP estimator is when the noise is white and gaussian, *i.e.*, $\boldsymbol{K}_{\boldsymbol{\varepsilon}} = \sigma^2\boldsymbol{I}$, and when one assumes that the object vector is zero mean ($\boldsymbol{\mu}_{\boldsymbol{x}} = \boldsymbol{0}$) and has independent and identically distributed components, *i.e.*, $\boldsymbol{K}_{\boldsymbol{x}} = \frac{1}{\beta}\boldsymbol{I}$. Under these (unrealistic) assumptions, the cost function is a simple combination of a least-squares **data-fit term** (negative log-likelihood) and an **energy penalty** term:

e,res,map,energy

$$\Psi(\boldsymbol{x}) = \frac{1}{2\sigma^2}\left\|\boldsymbol{y}-\boldsymbol{A}\boldsymbol{x}\right\|^2 + \frac{1}{2}\beta\left\|\boldsymbol{x}\right\|^2 \tag{1.7.9}$$

and the corresponding MAP estimator simplifies to:

e,res,map,gauss,I

$$\hat{\boldsymbol{x}}_{\text{MAP}} = \arg\min_{\boldsymbol{x}} \Psi(\boldsymbol{x}) = \left[\boldsymbol{A}'\boldsymbol{A} + \beta\sigma^2\boldsymbol{I}\right]^{-1} \boldsymbol{A}'\boldsymbol{y}. \tag{1.7.10}$$

In statistics, this estimation method is known as **ridge regression** [78] because the $\beta\boldsymbol{I}$ term reduces "ridges" in the cost function $\Psi(\boldsymbol{x})$. Such ridges are associated with **singular vectors** of $\boldsymbol{A}$ having small **singular values**. Chapter 15 discusses such estimators in more detail.

x,res,stat,gauss,white

Example 1.7.1 *Fig. 1.7.1 shows a random image $\boldsymbol{x}$, called a* ***gaussian random field*** *(**GRF**), drawn from the gaussian prior distribution (1.7.4) where $\boldsymbol{x} \sim \mathsf{N}(\boldsymbol{0}, \boldsymbol{I})$.*

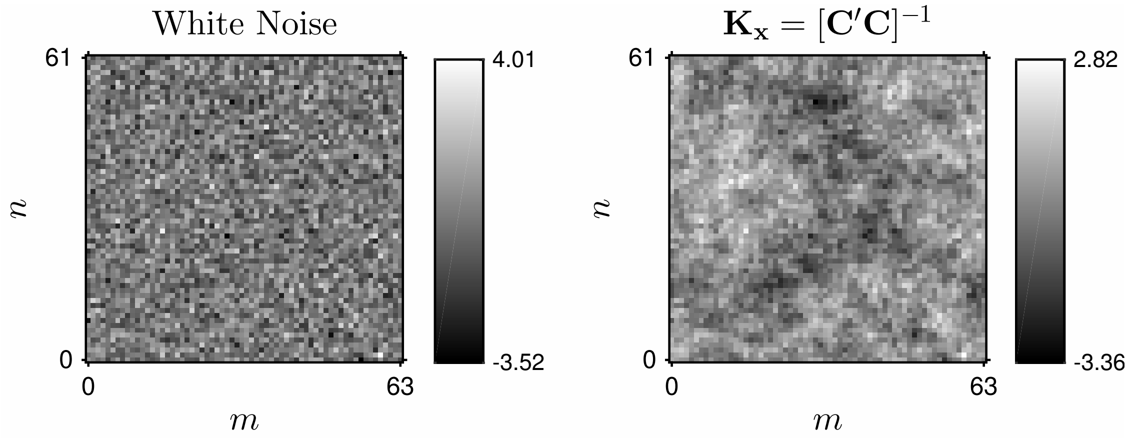


Figure 1.7.1: Random images drawn from the prior distribution for gaussian random fields (1.7.4) with $\boldsymbol{\mu}_{\boldsymbol{x}} = \boldsymbol{0}$. Left: $\boldsymbol{K}_{\boldsymbol{x}} = \boldsymbol{I}$. Right: $\boldsymbol{K}_{\boldsymbol{x}} = [\boldsymbol{C}'\boldsymbol{C}]^{-1}$ for the 2D differencing matrix $\boldsymbol{C}$ defined in (1.10.8).

fig_res_prior1

1.7.3.5 Circulant approximation in white gaussian case (Wiener filter)

The MAP estimator (1.7.10), involves a large matrix inverse that may provide little intuition. For insight into the MAP estimator's properties, consider the case where $\boldsymbol{A}$ is circulant, so $\boldsymbol{A} = \boldsymbol{Q}^{-1}\boldsymbol{\Gamma}\boldsymbol{Q}$, where $\boldsymbol{\Gamma} = \mathsf{diag}\{B_k\}$ and B_k was defined in (1.4.26). In the circulant case with white gaussian noise the MAP estimator (1.7.10), simplifies as follows:

$$\begin{aligned} \hat{\boldsymbol{x}}_{\text{MAP}} &= \left[(\boldsymbol{Q}^{-1}\boldsymbol{\Gamma}\boldsymbol{Q})'(\boldsymbol{Q}^{-1}\boldsymbol{\Gamma}\boldsymbol{Q}) + \beta\sigma^2\boldsymbol{I}\right]^{-1}(\boldsymbol{Q}^{-1}\boldsymbol{\Gamma}\boldsymbol{Q})'\boldsymbol{y} \\ &= \left[\boldsymbol{Q}^{-1}\boldsymbol{\Gamma}'\boldsymbol{\Gamma}\boldsymbol{Q} + \beta\sigma^2\boldsymbol{Q}^{-1}\boldsymbol{Q}\right]^{-1}\boldsymbol{Q}^{-1}\boldsymbol{\Gamma}'\boldsymbol{Q}\boldsymbol{y} \\ &= \boldsymbol{Q}^{-1}\left[\boldsymbol{\Gamma}'\boldsymbol{\Gamma} + \beta\sigma^2\boldsymbol{I}\right]^{-1}\boldsymbol{\Gamma}'\boldsymbol{Q}\boldsymbol{y} = \boldsymbol{Q}^{-1}\,\mathsf{diag}\{L_k\}\,\boldsymbol{Q}\boldsymbol{y}. \end{aligned} \tag{1.7.11}$$

This is just a filter having the following frequency response:

e,res,map,gauss,Lk

$$L_k = \frac{B_k^*}{|B_k|^2 + \beta\sigma^2}. \tag{1.7.12}$$

This is a DFT-based **Wiener filter** that has the following properties.

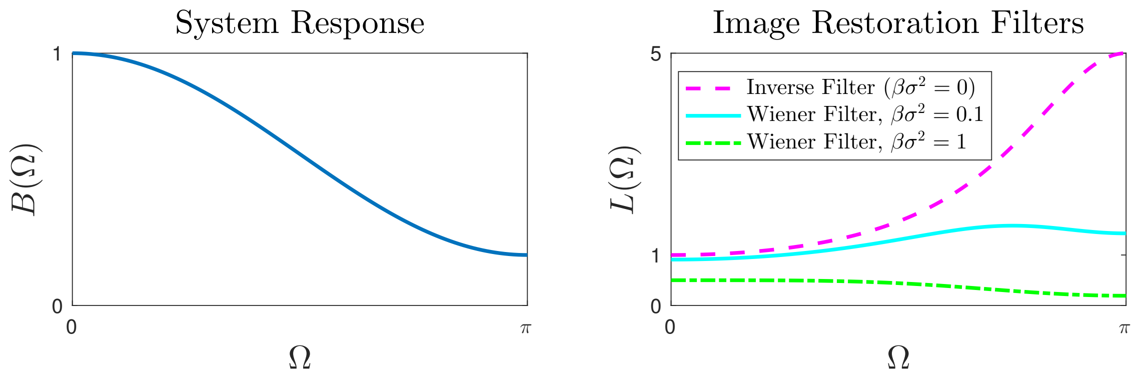
- As $\beta \to 0$, this filter approaches the inverse filter $1/B_k$.
- As $\beta \to \infty$, the frequency response of this filter approaches zero.

  A limit of zero is "sensible" because the assumed prior mean is $\boldsymbol{0}$; large β corresponds to high prior confidence in that mean, so the MMSE estimation strategy degenerates to the zero estimate.

Fig. 1.7.2 illustrates the properties of this filter for the case $b[n] = \frac{1}{5}\,\delta[n-1] + \frac{3}{5}\,\delta[n] + \frac{1}{5}\,\delta[n+1]$ for which the **DFT coefficients** are $B_k = B(2\pi k/N)$ where the corresponding **DTFT**, defined in (1.4.27), is $B(\Omega) = \frac{3}{5} + \frac{2}{5}\cos\Omega$ and $L_k = L(2\pi k/N)$.

Demo *See* `demo_res_wiener.m`.

Shrinking the entire estimate down to zero is unnatural for most imaging problems. So apparently the gaussian prior with mean zero and covariance $\boldsymbol{K}_{\boldsymbol{x}} = (1/\beta)\boldsymbol{I}$ is an unrealistic model for typical real-world images. Although one could certainly improve on this simplistic prior, it is nevertheless quite difficult in general to formulate realistic priors. Therefore, throughout the remainder of this book, we abandon the Bayesian philosophy and focus instead on a close cousin: **penalized-likelihood** estimators. The implementations of MAP methods and penalized-likelihood methods are quite similar, but the the philosophy and terminology differ somewhat. In the end it is something of a matter of personal preference.



fig_dft_wiener

Figure 1.7.2: Illustration of DFT-based Wiener filter frequency response $L(\Omega)$ for three values of $\beta\sigma^2$.

s,res,stat,pl

1.8 Penalized-likelihood estimation (s,res,stat,pl)

The general form of a MAP estimator (1.7.2) involves maximizing a function that consists of a log-likelihood term that quantifies agreement with the measurements, and another term that quantifies agreement with prior expectations. For the gaussian example (1.7.9) the log-prior was simply proportional to the negative of the **energy** of the signal, *i.e.*, $\log \mathsf{p}(\boldsymbol{x}) = -\beta \|\boldsymbol{x}\|^2$.

In the framework of **penalized-likelihood** estimation, one also finds the minimizer of a cost function consisting of two terms. Specifically, one finds $\hat{\boldsymbol{x}}$ by minimizing a cost function of the following form:

e,res,pl

$$\hat{\boldsymbol{x}} = \arg\min_{\boldsymbol{x}} \Psi(\boldsymbol{x})$$

$$\Psi(\boldsymbol{x}) = -\log \mathsf{p}(\boldsymbol{y} \,|\, \boldsymbol{x}) + \beta\, \mathsf{R}(\boldsymbol{x})\,. \tag{1.8.1}$$

The first term is the negative of the log-likelihood, which quantifies the disagreement between $\boldsymbol{x}$ and the measurements $\boldsymbol{y}$. We would like this term to be small. The second term is a regularizing **penalty function** $\mathsf{R} : \mathbb{R}^{n_{\mathrm{p}}} \to \mathbb{R}$ that penalizes an object $\boldsymbol{x}$ according to how much it departs from our assumptions about image properties.

The **regularization parameter** β controls the trade-off between the fit to the data and (typically) the smoothness of $\hat{\boldsymbol{x}}$. For very small β, $\hat{\boldsymbol{x}}$ will closely fit the data, which usually means very good spatial resolution in the absence of noise, but very noisy estimates in the presence of noise. Conversely, for large β, $\hat{\boldsymbol{x}}$ will emphasize minimizing $\mathsf{R}(\boldsymbol{x})$, which usually means a smooth estimate with low noise. Chapter 2 and Chapter 22 examine this trade-off in more detail.

Choosing the penalty $\mathsf{R}(\boldsymbol{x})$ is an art involving a multitude of trade-offs that are discussed further in Chapter 2. For example, if we expect objects to have small values, then the **energy** penalty

e,res,stat,pl,energy

$$\mathsf{R}(\boldsymbol{x}) = \frac{1}{2} \|\boldsymbol{x}\|^2 = \sum_{j=1}^{n_{\mathrm{p}}} \frac{1}{2} |x_j|^2 \tag{1.8.2}$$

could be reasonable. (This approach is also known as **Miller regularization** [79].) But in most imaging problems we expect the object to have nonzero energy, so this choice is not the best penalty function.

Why do we need a penalty function in the first place? Because the absence of such a term in the cost function leads again to the "inverse" solution (ML estimate) which greatly amplifies noise and leads to large image oscillations. These oscillations are contrary to our prior expectations about what images look like. So penalty functions that discourage highly oscillatory images usually are more natural than an energy penalty.

A penalty function that discourages high spatial frequency oscillations is called a **roughness penalty**. The simplest type of roughness penalty discourages disparities between neighboring pixel values. It is easiest to first see the effects of such a penalty function in a 1D example, as follows.

s,res,stat,1d

1.8.1 1D 1st-order roughness penalty

Consider a 1D problem where the elements of $\boldsymbol{x} \in \mathbb{R}^N$ correspond to consecutive "pixel" values $(f[0], \ldots, f[N-1])$. A natural measure of "roughness" of such a signal is the following penalty function that improves somewhat on (1.8.2):

e,res,pl,rough1

$$\mathsf{R}(\boldsymbol{x}) = \sum_{j=2}^{N} \frac{1}{2} |x_j - x_{j-1}|^2 = \sum_{n=1}^{N-1} \frac{1}{2} |f[n] - f[n-1]|^2 \,, \tag{1.8.3}$$

using (1.4.7). This penalty function assigns a high cost when neighboring pixel values are very different, thereby discouraging roughness and favoring spatially smooth estimates.

To proceed, it will be convenient to translate this penalty function into a matrix-vector form. Towards that end, consider the following $(N-1)\times N$ **finite differencing** matrix:

e,res,C,N-1,N

$$\boldsymbol{C} = \boldsymbol{D}_N \triangleq \begin{bmatrix} -1 & 1 & 0 & 0 & \dots & 0 \\ 0 & -1 & 1 & 0 & \dots & 0 \\ & & \ddots & \ddots & & \\ 0 & \dots & 0 & -1 & 1 & 0 \\ 0 & \dots & 0 & 0 & -1 & 1 \end{bmatrix}, \text{ so } \boldsymbol{C}\boldsymbol{x} = \begin{bmatrix} x_2 - x_1 \\ \vdots \\ x_N - x_{N-1} \end{bmatrix}. \tag{1.8.4}$$

This $\boldsymbol{C}$ is a **sparse matrix** because most of its elements are zero. (See §2.14.1.3.) With $\boldsymbol{C}$ thus defined, clearly $[\boldsymbol{C}\boldsymbol{x}]_k = x_{k+1} - x_k$, so

e,res,rough1,hess

$$\mathsf{R}(\boldsymbol{x}) = \sum_{k=1}^{N-1} \frac{1}{2} \left|[\boldsymbol{C}\boldsymbol{x}]_k\right|^2 = \frac{1}{2} \|\boldsymbol{C}\boldsymbol{x}\|^2 = \frac{1}{2}\boldsymbol{x}'\boldsymbol{C}'\boldsymbol{C}\boldsymbol{x} = \frac{1}{2}\boldsymbol{x}'\boldsymbol{\mathsf{R}}\boldsymbol{x}. \tag{1.8.5}$$

The matrix $\boldsymbol{\mathsf{R}}$ is the **Hessian** of the roughness penalty $\mathsf{R}(\boldsymbol{x})$, *i.e.*, the matrix of its second partial derivatives [wiki]. Here, $\boldsymbol{\mathsf{R}}$ is the following "nearly **Toeplitz**" matrix:

e,res,rough1,R

$$\boldsymbol{\mathsf{R}} \triangleq \boldsymbol{C}'\boldsymbol{C} = \begin{bmatrix} 1 & -1 & 0 & \dots & 0 & 0 \\ -1 & 2 & -1 & 0 & \dots & 0 \\ & & & & & \\ & & \ddots & \ddots & \ddots & \\ & & & & & \\ 0 & \dots & 0 & -1 & 2 & -1 \\ 0 & 0 & \dots & 0 & -1 & 1 \end{bmatrix}. \tag{1.8.6}$$

This type of penalty function is called a **quadratic penalty** because $\mathsf{R}(\boldsymbol{x})$ is a quadratic form in $\boldsymbol{x}$. The quadratic form is particularly convenient when the measurements have gaussian distributions. See [80] for analysis of the eigenvalues and eigenvectors of (1.8.6) for various boundary conditions including **Neumann boundary conditions** and **Dirichlet boundary conditions**.

Often it is expected that $f[n]$ is zero near the ends of its support, such as in 2D tomography problems where there is usually "black air space" surrounding the body. In such case, an alternate penalty function is

$$\mathsf{R}(\boldsymbol{x}) = \frac{1}{2}\left|f[0] - 0\right|^2 + \frac{1}{2}\left|f[N-1] - 0\right|^2 + \sum_{n=1}^{N-1} \frac{1}{2}\left|f[n] - f[n-1]\right|^2.$$

In this case, the form $\mathsf{R}(\boldsymbol{x}) = \frac{1}{2}\|\boldsymbol{C}\boldsymbol{x}\|^2$ in (1.8.5) still applies, but here the $(N+1)\times N$ differencing matrix $\boldsymbol{C}$ has the following form:

e,res,C,N+1,N

$$\boldsymbol{C} = \left[\begin{array}{ccccc} 1 & 0 & \dots & 0 & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ & & \boldsymbol{D}_N & & \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & \dots & 0 & 1 \end{array}\right], \tag{1.8.7}$$

where $\boldsymbol{D}_N$ was defined in (1.8.4). The Hessian $\boldsymbol{\mathsf{R}} = \boldsymbol{C}'\boldsymbol{C}$ for this penalty is *exactly* **Toeplitz** with elements

$$R_{kj} = 2\,\delta[k-j] - \delta[k-j-1] - \delta[k-j+1].$$

Having a constant diagonal simplifies slightly the implementation of some iterative algorithms.

s,res,stat,pl,gauss

1.8.2 Linear gaussian case: QPWLS estimator

Combining the log-likelihood corresponding to the linear gaussian case (1.7.3) with the preceding quadratic penalty function (1.8.5) yields the following cost function for penalized-likelihood estimation:

e,res,gauss,pl,Kx

$$\Psi(\boldsymbol{x}) = \frac{1}{2}(\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x})'\boldsymbol{K}_{\boldsymbol{\varepsilon}}^{-1}(\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}) + \beta\frac{1}{2}\boldsymbol{x}'\boldsymbol{\mathsf{R}}\boldsymbol{x}. \tag{1.8.8}$$

By similar analysis as in §1.7.3.1, the minimizer[11] of $\Psi(\boldsymbol{x})$ is

e,res,xh,qpwls

$$\hat{\boldsymbol{x}} = \left[\boldsymbol{A}'\boldsymbol{K}_{\boldsymbol{\varepsilon}}^{-1}\boldsymbol{A} + \beta\boldsymbol{\mathsf{R}}\right]^{-1}\boldsymbol{A}'\boldsymbol{K}_{\boldsymbol{\varepsilon}}^{-1}\boldsymbol{y}. \tag{1.8.9}$$

[11] One must address the invertibility of $\boldsymbol{A}'\boldsymbol{K}_{\boldsymbol{\varepsilon}}^{-1}\boldsymbol{A} + \beta\boldsymbol{\mathsf{R}}$ in (1.8.9), because in general $\boldsymbol{A}'\boldsymbol{K}_{\boldsymbol{\varepsilon}}^{-1}\boldsymbol{A}$ and $\boldsymbol{\mathsf{R}}$ are each only positive semidefinite. For penalty functions based on first-order differences, like (1.8.3) and (1.10.1), the null space of $\boldsymbol{\mathsf{R}}$ is spanned by $\boldsymbol{1}$, the vector of all ones. For most imaging systems, $\boldsymbol{A}\boldsymbol{1} \neq \boldsymbol{0}$, so the null spaces of $\boldsymbol{A}'\boldsymbol{K}_{\boldsymbol{\varepsilon}}^{-1}\boldsymbol{A}$ and $\boldsymbol{\mathsf{R}}$ are disjoint. For such systems, their sum is invertible.

This regularized solution dates back at least to [81, eqn. (7)] and it is simply a special case of the MAP estimator (1.7.6), where $\boldsymbol{\mu}_{\boldsymbol{x}} = \mathbf{0}$ and $\beta\mathbf{R} = \boldsymbol{K}_{\boldsymbol{x}}^{-1}$. In Bayesian language, the penalty function (1.8.3) corresponds to the following improper gaussian prior:

e,res,stat,pl,prior

$$\mathsf{p}(\boldsymbol{x}) = c\,\mathrm{e}^{-\beta\frac{1}{2}\boldsymbol{x}'\mathbf{R}\boldsymbol{x}}\,. \tag{1.8.10}$$

This is an **improper prior** because the $\mathbf{R}$ in (1.8.6) above is not invertible. Indeed most useful quadratic penalty functions have singular Hessians, a property that is entirely acceptable in penalized-likelihood estimation but is perhaps somewhat awkward for a **Bayesian** interpretation.

We refer to (1.8.8) and (1.8.9) as a quadratically penalized weighted least-squares (**QPWLS**) estimator.

s,res,stat,pl,circ

1.8.3 Circulant analysis of QPWLS restoration

For further insight into the properties of the QPWLS estimator (1.8.9), it is again convenient to apply circulant approximations.

First we observe that in typical tomographic imaging applications the object usually does not fill the entire field of view (think of the "black air space" around the head in a brain scan). So we alter the form of $\mathsf{R}(\boldsymbol{x})$ to make it a circular shift-invariant functional by analyzing the following roughness penalty function:

e,res,stat,pl,circ,Rx

$$\mathsf{R}(\boldsymbol{x}) = \frac{1}{2}\left|x_1 - x_N\right|^2 + \sum_{j=2}^{N} \frac{1}{2}\left|x_j - x_{j-1}\right|^2. \tag{1.8.11}$$

The initial term is needed for analysis only and is rarely implemented in practice.

To put this penalty function in matrix-vector form, consider the following $N \times N$ modified $\boldsymbol{C}$ matrix:

e,res,stat,pl,C,circ

$$\boldsymbol{C} = \begin{bmatrix} -1 & 1 & 0 & \dots & 0 \\ 0 & -1 & 1 & \dots & 0 \\ & & \ddots & \ddots & \\ 0 & \dots & 0 & -1 & 1 \\ 1 & 0 & \dots & 0 & -1 \end{bmatrix}, \text{ where } \boldsymbol{C}\boldsymbol{x} = \begin{bmatrix} x_2 - x_1 \\ \vdots \\ x_N - x_{N-1} \\ x_1 - x_N \end{bmatrix}. \tag{1.8.12}$$

This $\boldsymbol{C}$ matrix again satisfies (1.8.5) for the preceding $\mathsf{R}(\boldsymbol{x})$, where now $\mathbf{R}$ is the following **circulant matrix**[12]

e,res,stat,pl,R

$$\mathbf{R} \triangleq \boldsymbol{C}'\boldsymbol{C} = \begin{bmatrix} 2 & -1 & 0 & \dots & 0 & -1 \\ -1 & 2 & -1 & 0 & \dots & 0 \\ & & \ddots & \ddots & \ddots & \\ 0 & \dots & 0 & -1 & 2 & -1 \\ -1 & 0 & \dots & 0 & -1 & 2 \end{bmatrix}. \tag{1.8.13}$$

Because $\mathbf{R}$ is circulant, it is diagonalized by the DFT matrix $\boldsymbol{Q}^{-1}$, *i.e.*, $\mathbf{R} = \boldsymbol{Q}^{-1}\boldsymbol{\Pi}\boldsymbol{Q}$ where $\boldsymbol{\Pi}$ is diagonal with entries R_k. In particular, evaluating the DFT of the first column of this $\mathbf{R}$ shows that its eigenvalues (DFT coefficients) are[13]

e,res,stat,pl,eRk

$$\mathsf{R}_k = 2 - \mathrm{e}^{-\imath 2\pi k/N} - \mathrm{e}^{\imath 2\pi k/N} = 2 - 2\cos(2\pi k/N)\,. \tag{1.8.14}$$

Suppose further that the system matrix $\boldsymbol{A}$ is also circulant, *i.e.*, $\boldsymbol{A} = \boldsymbol{Q}^{-1}\boldsymbol{\Gamma}\boldsymbol{Q}$, and that the noise is white: $\boldsymbol{K}_{\boldsymbol{\varepsilon}} = \sigma^2\boldsymbol{I}$. Substituting these assumptions into the QPWLS estimator (1.8.9) yields

$$\begin{aligned} \hat{\boldsymbol{x}} = \arg\min_{\boldsymbol{x}} \Psi(\boldsymbol{x}) &= \left[\boldsymbol{A}'\boldsymbol{A} + \beta\sigma^2\mathbf{R}\right]^{-1}\boldsymbol{A}'\boldsymbol{y} \\ &= \boldsymbol{Q}^{-1}\left[\boldsymbol{\Gamma}'\boldsymbol{\Gamma} + \beta\sigma^2\boldsymbol{\Pi}\right]^{-1}\boldsymbol{\Gamma}'\boldsymbol{Q}\boldsymbol{y} = \boldsymbol{Q}^{-1}\,\mathsf{diag}\{L_k\}\,\boldsymbol{Q}\boldsymbol{y}. \end{aligned} \tag{1.8.15}$$

So once again our estimator $\hat{\boldsymbol{x}}$ is a linear, circularly shift-invariant, Wiener-like filter[14] with frequency response

e,res,wiener,Lk

$$L_k = \frac{B_k^*}{\left|B_k\right|^2 + \beta\sigma^2\mathsf{R}_k}. \tag{1.8.16}$$

Again, as $\beta \to 0$, this filter approaches the inverse filter: $L_k \to 1/B_k$. This property is inherent to the form (1.8.1). However, because $R_0 = 0$, as $\beta \to \infty$, we have the following:

$$L_k \to \begin{cases} 1/B_0, & k = 0 \\ 0, & \text{otherwise.} \end{cases}$$

[12] To realize how special this matrix is, see http://www.siam.org/news/news.php?id=1697.

f,eig,R

[13] For the special case of the first-order roughness penalty (1.8.3), one can determine its eigenvalues and eigenvectors exactly without imposing the circulant model, *e.g.*, [82]. The eigenvalues of (1.8.6) are $\mathsf{R}_k = 2 - 2\cos(\pi k/N),\ k = 0, \dots, N-1$. Nevertheless, the circulant approximation provides more intuition because of its frequency response interpretation.

[14] Technically speaking the term **Wiener filter** is a Bayesian concept because it was derived originally using a prior model for the object $\boldsymbol{x}$. In image restoration, the traditional non-Bayesian term for this approach is **constrained least-squares** [83]. Here we treat $\mathsf{R}(\boldsymbol{x})$ as a penalty function rather than as a constraint.

This behavior is preferable to that of the energy penalty (1.8.2) because at least the DC term ($k = 0$) is unaffected by the regularizing penalty function (1.8.11).

x,res,stat,lpl,gauss

Example 1.8.1 *Fig. 1.8.1 illustrates these properties for the same example considered in Fig. 1.7.2. As desired, the QPWLS filter response at low frequencies is nearly unity here, unlike in Fig. 1.7.2.*

Mat *See* `deconvreg` *and* `deconvwnr`.

Demo *See* `demo_res_wiener.m`.

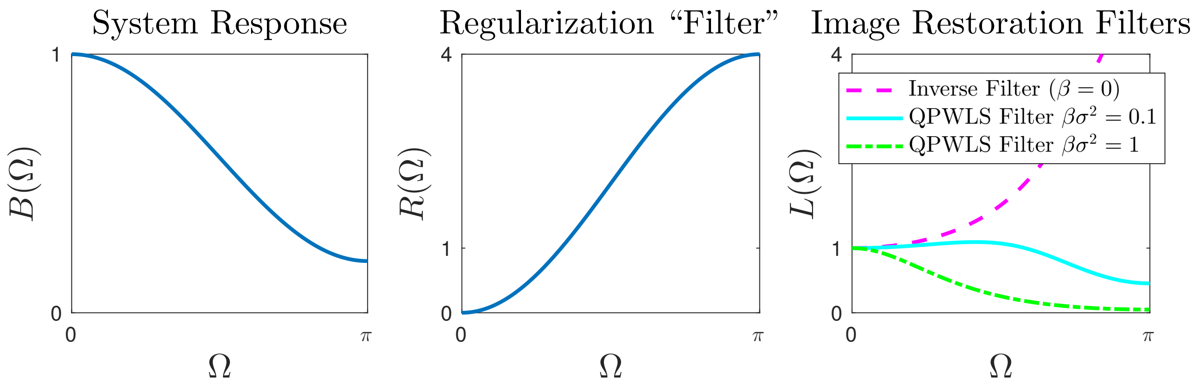


Figure 1.8.1: Illustration of properties of the Wiener-like filter (1.8.16) corresponding to QPWLS estimation for 1D signal restoration example. fig_dft_wiener_R

1.8.4 Discussion

Although the roughness penalty function (1.8.3) yields behavior that is preferable to the energy penalty function (1.8.2), it still yields a linear method that is essentially a lowpass filter, so we can reduce noise only by compromising spatial resolution. This is a fundamental **resolution-noise trade-off** in all (linear) imaging systems and image processing methods. This trade-off can be overcome only by considering nonlinear estimators. Nonquadratic penalty functions can smooth noise while preserving image edges so are a focus of the algorithms discussed in this *book*. However, being nonlinear they are less amenable to the types of "signal processing" analysis described above.

As discussed in Chapter 22, in certain imaging problems it can be desirable to formulate a penalty function $\mathsf{R}(\boldsymbol{x})$ that *depends on the measurements* $\boldsymbol{y}$, *e.g.*, [84]–[86]. Such formulations are perfectly compatible with the penalized-likelihood philosophy, but seem at odds with the Bayesian notion of a "prior." This is another reason why we prefer the former philosophy. [RQ]

s,res,pl,mav1

1.9 Mean and variance analysis (resolution-noise trade-offs) (s,res,pl,mav1)

For further insight into the **resolution-noise trade-off** associated with penalized-likelihood estimation, this section analyzes the first two moments of the 1D linear QPWLS estimator $\hat{\boldsymbol{x}}$ in (1.8.15). Chapter 22 and Chapter 23 consider more complicated nonlinear problems. Ideally we would analyze the entire statistical behavior of an estimator $\hat{\boldsymbol{x}}$ by studying its probability distribution $\mathsf{p}(\hat{\boldsymbol{x}})$. Unfortunately, for nonlinear estimators such analysis is intractable except in some special cases, *e.g.*, [87]. Therefore, we focus on the first two moments of $\hat{\boldsymbol{x}}$, or, in nonlinear cases, the approximations thereof.

The QPWLS estimator $\hat{\boldsymbol{x}}$ in (1.8.15) is linear, and we can write it as $\hat{\boldsymbol{x}} = \boldsymbol{L}\boldsymbol{y}$ where

$$\boldsymbol{L} \triangleq \left[\boldsymbol{A}'\boldsymbol{K}_{\boldsymbol{\varepsilon}}^{-1}\boldsymbol{A} + \beta\mathbf{R}\right]^{-1}\boldsymbol{A}'\boldsymbol{K}_{\boldsymbol{\varepsilon}}^{-1} = \boldsymbol{Q}^{-1}\left[\boldsymbol{\Gamma}'\boldsymbol{\Gamma} + \beta\sigma^2\boldsymbol{\Pi}\right]^{-1}\boldsymbol{\Gamma}'\boldsymbol{Q}.$$

Recall that $\boldsymbol{y} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{\varepsilon}$ where $\boldsymbol{\varepsilon} \sim \mathsf{N}\big(\boldsymbol{0}, \sigma^2\boldsymbol{I}\big)$, so $\mathsf{E}[\boldsymbol{y} \,|\, \boldsymbol{x}] = \boldsymbol{A}\boldsymbol{x}$ and $\boldsymbol{K}_{\boldsymbol{\varepsilon}} = \mathsf{Cov}\{\boldsymbol{y} \,|\, \boldsymbol{x}\} = \sigma^2\boldsymbol{I}$. Thus, by the linearity of expectation, the mean of $\hat{\boldsymbol{x}}$ for a given $\boldsymbol{x}$ is

$$\begin{aligned}\mathsf{E}[\hat{\boldsymbol{x}} \,|\, \boldsymbol{x}] &= \mathsf{E}[\boldsymbol{L}\boldsymbol{y} \,|\, \boldsymbol{x}] = \boldsymbol{L}\,\mathsf{E}[\boldsymbol{y} \,|\, \boldsymbol{x}] = \boldsymbol{L}\boldsymbol{A}\boldsymbol{x} = \left[\boldsymbol{A}'\boldsymbol{A} + \beta\sigma^2\mathbf{R}\right]^{-1}\boldsymbol{A}'\boldsymbol{A}\boldsymbol{x} \\ &= \boldsymbol{Q}^{-1}\left[\boldsymbol{\Gamma}'\boldsymbol{\Gamma} + \beta\sigma^2\boldsymbol{\Pi}\right]^{-1}\boldsymbol{\Gamma}'\boldsymbol{\Gamma}\boldsymbol{Q}\boldsymbol{x} = \boldsymbol{Q}^{-1}\,\mathsf{diag}\{M_k\}\,\boldsymbol{Q}\boldsymbol{x}. \qquad (1.9.1)\end{aligned}$$

So the expectation of $\hat{\boldsymbol{x}}$ is simply a filtered version of the true object $\boldsymbol{x}$ with filter frequency response

e,res,pl,mean,Mk

$$M_k = \frac{|B_k|^2}{|B_k|^2 + \beta\sigma^2\mathsf{R}_k} = L_k B_k, \qquad (1.9.2)$$

because $\boldsymbol{\Gamma} = \mathsf{diag}\{B_k\}$ and $\mathbf{\Pi} = \mathsf{diag}\{\mathsf{R}_k\}$, where L_k was defined in (1.8.16). For good spatial resolution, we would like this "filter" to pass all spatial frequencies with a gain of unity, which means we would like $\beta \mathsf{R}_k$ to be small.

To analyze the covariance of $\hat{\boldsymbol{x}}$, we use the following particularly important property of covariance matrices:

e,cov(Ax)

$$\mathsf{Cov}\{\boldsymbol{L}\boldsymbol{z}\} = \boldsymbol{L}\,\mathsf{Cov}\{\boldsymbol{z}\}\,\boldsymbol{L}', \tag{1.9.3}$$

which follows directly from the definition (1.6.2). Thus the covariance of $\hat{\boldsymbol{x}}$ for a given true object $\boldsymbol{x}$ is:

$$\begin{aligned}\mathsf{Cov}\{\hat{\boldsymbol{x}}\,|\,\boldsymbol{x}\} &= \mathsf{Cov}\{\boldsymbol{L}\boldsymbol{y}\,|\,\boldsymbol{x}\} = \boldsymbol{L}\,\mathsf{Cov}\{\boldsymbol{y}\,|\,\boldsymbol{x}\}\,\boldsymbol{L}' = \boldsymbol{L}(\sigma^2 \boldsymbol{I})\boldsymbol{L}' \\ &= \sigma^2 \left[\boldsymbol{A}'\boldsymbol{A} + \beta\sigma^2 \mathbf{R}\right]^{-1} \boldsymbol{A}'\boldsymbol{A} \left[\boldsymbol{A}'\boldsymbol{A} + \beta\sigma^2 \mathbf{R}\right]^{-1}. \end{aligned} \tag{1.9.4}$$

This matrix expression may not be particularly intuitive, so again we consider the circulant model (1.4.30). Then the covariance simplifies as follows:

$$\begin{aligned}\mathsf{Cov}\{\hat{\boldsymbol{x}}\,|\,\boldsymbol{x}\} &= \sigma^2 \boldsymbol{Q}^{-1} \left[\boldsymbol{\Gamma}'\boldsymbol{\Gamma} + \beta\sigma^2 \mathbf{\Pi}\right]^{-1} \boldsymbol{\Gamma}'\boldsymbol{\Gamma} \left[\boldsymbol{\Gamma}'\boldsymbol{\Gamma} + \beta\sigma^2 \mathbf{\Pi}\right]^{-1} \boldsymbol{Q} \\ &= \boldsymbol{Q}^{-1}\,\mathsf{diag}\{P_k\}\,\boldsymbol{Q}, \end{aligned} \tag{1.9.5}$$

where the following diagonal entries correspond to the **noise power spectrum** (**NPS**) of the estimator $\hat{\boldsymbol{x}}$:

e,res,pl,nps

$$P_k \triangleq \frac{\sigma^2 |B_k|^2}{\left(|B_k|^2 + \beta\sigma^2 \mathsf{R}_k\right)^2} = \sigma^2 |L_k|^2 . \tag{1.9.6}$$

For a low-noise restoration, we would like the variance of $\hat{x}_j$ to be small. The variance of $\hat{x}_j$ is given by

$$\begin{aligned}\mathsf{Var}\{\hat{x}_j\,|\,\boldsymbol{x}\} &= \boldsymbol{e}_j'\,\mathsf{Cov}\{\hat{\boldsymbol{x}}\,|\,\boldsymbol{x}\}\,\boldsymbol{e}_j = \boldsymbol{e}_j'\boldsymbol{Q}^{-1}\,\mathsf{diag}\{P_k\}\,\boldsymbol{Q}\boldsymbol{e}_j \\ &= \frac{1}{N}\sum_{k=0}^{N-1} P_k = \frac{\sigma^2}{N}\sum_{k=0}^{N-1} |L_k|^2 , \end{aligned} \tag{1.9.7}$$

because the DFT of a Kronecker impulse is unity, *i.e.*, $|\boldsymbol{Q}\boldsymbol{e}_j|_k = 1$, and we use (1.4.29). In 2D the factor $\frac{\sigma^2}{N}$ is $\frac{\sigma^2}{MN}$. Reducing image noise is equivalent to having a low variance. From (1.9.7), having a small variance requires $\beta \mathsf{R}_k$ to be large whenever B_k is small.

Comparing (1.9.2) and (1.9.7) illustrates succinctly the fundamental **resolution-noise trade-off** in any linear image restoration method. For good spatial resolution, we want (1.9.2) to be approximately unity, which means $\beta \mathsf{R}_k \approx 0$, but for low noise, (1.9.7) requires that $\beta \mathsf{R}_k$ be large.

As a concrete example, consider the **image denoising** problem where the system impulse response in (1.2.3) is $b[n] = \delta[n]$, so $B_k = 1$. If $\beta = 1$ and $\sigma^2 = 1$, then (1.9.2) gives $\frac{1}{1+R}$ whereas (1.9.7) gives $\frac{1}{(1+R)^2}$. We cannot simultaneously make the first equation near unity and the second equation near 0. Consequently, if we use a nonzero R to reduce noise, we will also reduce high spatial frequencies. The principal visual effect of this attenuation will be blurring of edges in the image.

The origin of this undesirable trade-off is the quadratic roughness penalty function itself. In Bayesian terms, a quadratic roughness penalty corresponds to a gaussian prior, yet most real world images are not particularly gaussian. To overcome this limitation we will consider nonlinear methods based on nonquadratic penalty functions. But first we consider how to form roughness penalties for 2D images.

Fig. 1.9.1 illustrates this trade-off by showing the overall frequency response M_k in (1.9.2) and noise power spectrum P_k in (1.9.6) for three values of the regularization parameter β. As β decreases, M_k improves towards unity, but P_k increases.

Demo *See* `demo_res_wiener.m`. [RQ]

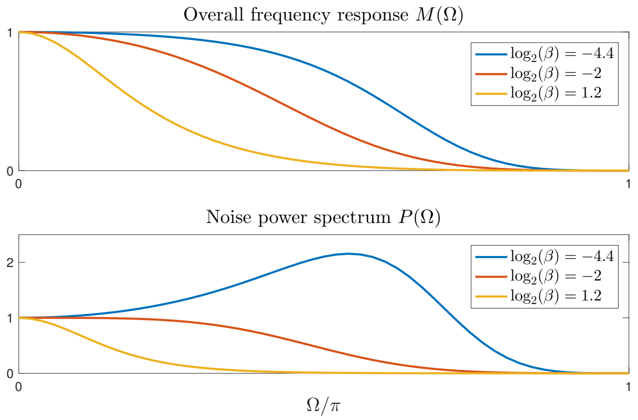


Figure 1.9.1: Illustration of trade-off between spatial resolution (over all frequency response) and noise power spectrum in image restoration with a quadratic roughness penalty, for three values of the regularization parameter β. fig_res_wiener_nps

s,res,penal2

1.10 Roughness penalties in 2D (s,res,penal2)

Thus far we have considered only the simple 1D roughness penalty of the form (1.8.3). Applying penalized-likelihood methods to image restoration problems requires 2D roughness penalty functions.

1.10.1 Quadratic regularization

Extrapolating (1.8.3) from 1D signals to 2D objects $f[m,n]$ suggests the following simple roughness penalty:

e,res,penal2,Rf,hv

$$\mathsf{R}(f) = \sum_{m=1}^{M-1} \sum_{n=0}^{N-1} \frac{1}{2} \left| f[m,n] - f[m-1,n] \right|^2 + \sum_{m=0}^{M-1} \sum_{n=1}^{N-1} \frac{1}{2} \left| f[m,n] - f[m,n-1] \right|^2 . \qquad (1.10.1)$$

This quadratic penalty function discourages disparities between horizontal and vertical neighboring pixels. For 2D roughness penalty functions, it is often useful to include *diagonal* neighboring pixels, *i.e.*, to include terms of the form $|f[m,n] - f[m-1,n-1]|^2$ for example. See §2.3 for further generalizations.

The notation in (1.10.1) becomes even more cumbersome in 3D, particularly as one uses more neighbors. For a more concise expression, write the penalty function in terms of the vector $\boldsymbol{x}$ instead of the 2D array $f[m,n]$, recalling (1.4.16), as follows:

e,res,penal2,Rx

$$\mathsf{R}(\boldsymbol{x}) = \sum_{j=1}^{n_{\mathrm{p}}} \sum_{l \in \mathcal{N}_j} \frac{1}{2} |x_j - x_l|^2 , \qquad (1.10.2)$$

where $\mathcal{N}_j$ denotes the set of neighbors of the jth pixel, or more precisely, half of that set of neighbors. For a $M \times N$ object, a **first-order neighborhood** means[15]

e,res,penal2,Nj,1

$$\mathcal{N}_j = \{j-1,\, j-M\} \qquad (1.10.3)$$

and a **2nd-order neighborhood** means

$$\mathcal{N}_j = \{j-1,\, j-M,\, j-M-1,\, j-M+1\} .$$

To help understand these sets, the following diagram illustrates the distinction between 2D pixel coordinates and 1D vector element indexes of lexicographically ordered arrays for a 3 × 3 neighborhood in a $M \times N$ image.

e,res,penal2,j,m,n

1D indexes

$j-M-1$	$j-M$	$j-M+1$
$j-1$	j	$j+1$
$j+M-1$	$j+M$	$j+M+1$

2D coordinates

$[m-1,n-1]$	$[m,n-1]$	$[m+1,n-1]$
$[m-1,n]$	$[m,n]$	$[m+1,n]$
$[m-1,n+1]$	$[m,n+1]$	$[m+1,n+1]$

$\rightarrow m$ $\downarrow n$ (1.10.4)

Again, an appropriate matrix-vector representation can further greatly simplify notation. Each term in (1.10.1) or (1.10.2) involves a difference of nearby pixel values, *e.g.*, $x_j - x_l$, which is a simple linear combination. For a first-order neighborhood, the 2D roughness penalty (1.10.2) has the following concise matrix-vector form:

e,res,penal2,Rx,hv

$$\mathsf{R}(\boldsymbol{x}) = \frac{1}{2} \|\boldsymbol{C}_1 \boldsymbol{x}\|^2 + \frac{1}{2} \|\boldsymbol{C}_2 \boldsymbol{x}\|^2 , \qquad (1.10.5)$$

provided we define appropriately the matrices $\boldsymbol{C}_1$ and $\boldsymbol{C}_2$. Each row of $\boldsymbol{C}_1$ corresponds to one term in the first summation in (1.10.1). The natural choice for $\boldsymbol{C}_1$ would have size $(M-1)N \times MN$, because this is the number of terms in the first sum in (1.10.1). However, it can be more convenient for implementation to choose $\boldsymbol{C}_1$ and $\boldsymbol{C}_2$ to both have size $MN \times MN$, allowing each matrix to have a few rows that are entirely zero. Such zero rows do not change the value of the penalty function. (Instead of being entirely zero, those rows could have entries that correspond to other end conditions.) See Fig. 1.10.1 and Problem 1.14.

Recalling (1.4.16), we can identify the term $f[m,n] - f[m-1,n]$ with the kth row of $\boldsymbol{C}_1$, where $k = 1+m+nM$. With this natural ordering, the elements of $\boldsymbol{C}_1$ are as follows:

e,res,penal2,C1kj

$$[\boldsymbol{C}_1]_{kj} = \begin{cases} 1, & k = j = 1+m+nM \\ -1, & k = 1+m+nM,\ j = k-1 \\ 0, & \text{otherwise}, \end{cases} \qquad (1.10.6)$$

for $m = 1,\ldots,M-1,\ n = 0,\ldots,N-1$, *i.e.*, for $k = 2,\ldots,n_{\mathrm{p}} = MN$. Each nonzero row of $\boldsymbol{C}_1$ has a single "-1" entry and a "1" entry, and all other elements are zero. Thus $\boldsymbol{C}_1$ is a very **sparse matrix**. One can define $\boldsymbol{C}_2$ similarly. (See §2.3.) One can verify that if $\boldsymbol{x}$ denotes the lexicographic representation of $f[m,n]$ per (1.4.14), then

$$[\boldsymbol{C}_1 \boldsymbol{x}]_k \Big|_{k=1+m+nM} = \begin{cases} f[m,n] - f[m-1,n], & \begin{matrix} m = 1,\ldots,M-1 \\ n = 0,\ldots,N-1 \end{matrix} \\ 0, & \quad \text{otherwise}. \end{cases}$$

[15] (1.10.3) must be modified slightly when the jth pixel is on the left or top border of the image.

We can write the quadratic roughness penalty (1.10.5) even more concisely as follows:

e,res,penal2,Rq

$$\mathsf{R}(\boldsymbol{x}) = \frac{1}{2} \left\| \boldsymbol{C} \boldsymbol{x} \right\|^2, \tag{1.10.7}$$

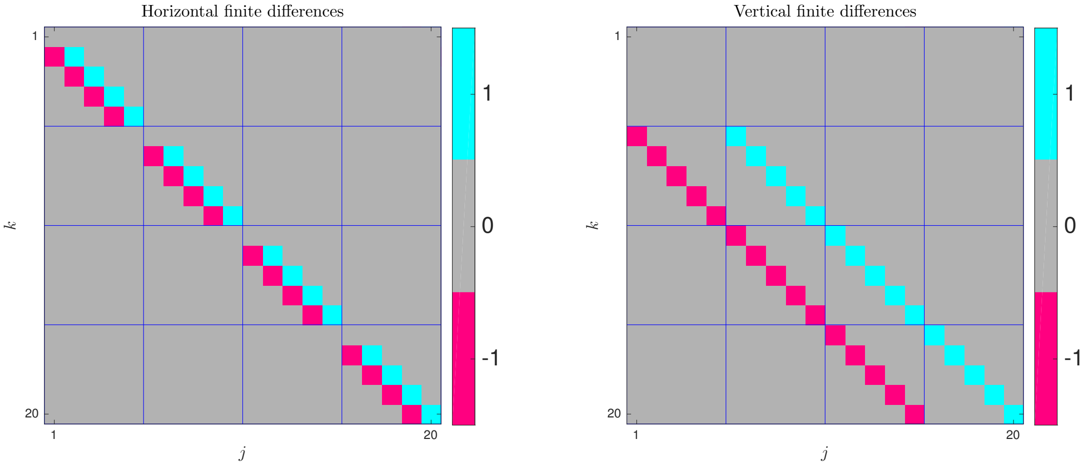
by defining the following $2MN \times MN$ matrix

e,res,penal2,C

$$\boldsymbol{C} = \begin{bmatrix} \boldsymbol{C}_1 \\ \boldsymbol{C}_2 \end{bmatrix}. \tag{1.10.8}$$

See Chapter 2 for extensions and details about implementing such $\boldsymbol{C}$ matrices.

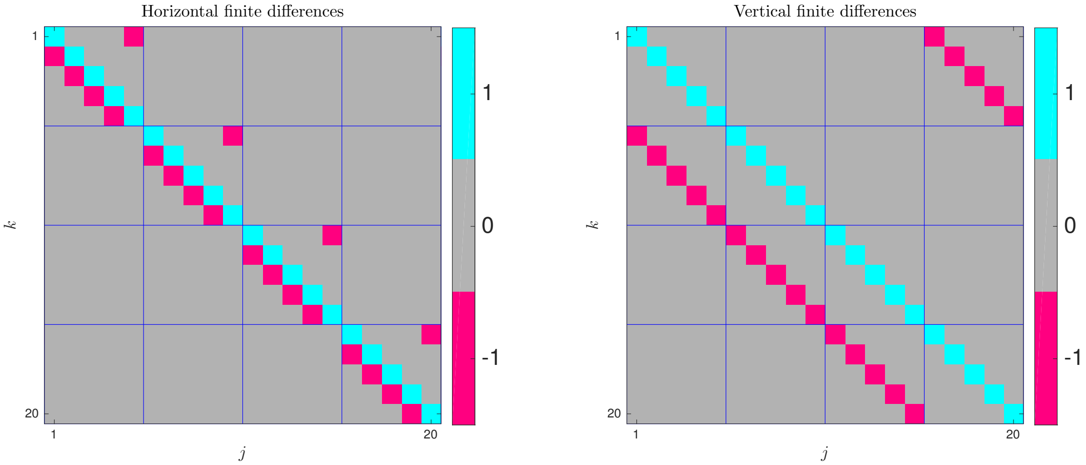
Fig. 1.10.1 illustrates $\boldsymbol{C}_1$ and $\boldsymbol{C}_2$ in (1.10.8) for a small 2D image. Each matrix has $N \times N$ blocks of $M \times M$ elements. Likewise, Fig. 1.10.2 shows the case where we use periodic boundary conditions. In this latter case, both $\boldsymbol{C}_1$ and $\boldsymbol{C}_2$ are **block circulant with circulant blocks** (**BCCB**).



fig_res_c2v

Figure 1.10.1: Illustration of finite-differencing matrices $\boldsymbol{C}_1$ and $\boldsymbol{C}_2$ for a 2D image with $M = 5$ and $N = 4$. There are $N \times N$ blocks each of size $M \times M$.

fig_res_c2h



fig_res_c2vc

Figure 1.10.2: Illustration of finite-differencing matrices $\boldsymbol{C}_1$ and $\boldsymbol{C}_2$ for a 2D image with $M = 5$ and $N = 4$ for the case of periodic boundary conditions.

fig_res_c2hc

The Hessian of the quadratic regularizer (1.10.7) is $\mathbf{R} = \nabla^2 \mathsf{R} = \boldsymbol{C}_1' \boldsymbol{C}_1 + \boldsymbol{C}_2' \boldsymbol{C}_2$. Fig. 1.10.3 illustrates this Hessian for the case of periodic boundary conditions, for which $\mathbf{R}$ is BCCB. Following the analysis in §1.4.3.2, the eigenvalues of this $\mathbf{R}$ correspond to the 2D DFT of the (suitably reshaped) first column of $\mathbf{R}$. That first column, when reshaped, corresponds to the impulse response

$$r[m,n] = 4\,\delta[m]\,\delta[n] - \delta[m-1]\,\delta[n] - \delta[m+1]\,\delta[n] - \delta[m]\,\delta[n-1] - \delta[m]\,\delta[n+1]$$

which has corresponding 2D DFT

$$\left\{ 4 - 2\cos\left(\frac{2\pi}{M} k\right) - 2\cos\left(\frac{2\pi}{N} l\right) \; : k = 0, \ldots, M-1, \; l = 0, \ldots, N-1 \right\},$$

and these are the MN eigenvalues of $\mathbf{R}$. See Problem 1.16.

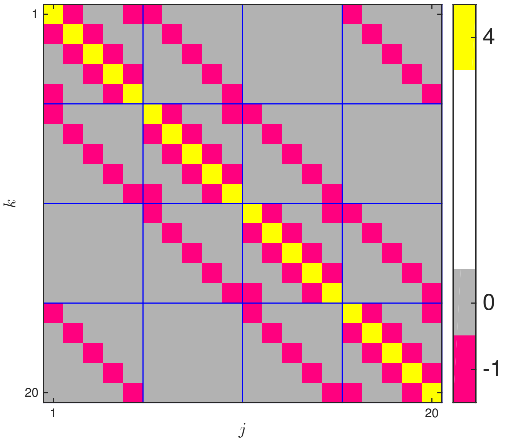


Figure 1.10.3: Illustration of quadratic regularizer Hessian $\mathsf{R} = C_1' C_1 + C_2' C_2$ for a 2D image with $M = 5$ and $N = 4$ for the case of periodic boundary conditions. This Hessian matrix is **BCCB**. fig_res_c2c_r

x,res,penal2,prior

Example 1.10.1 *As discussed in §1.8.2, using a penalty function of the form (1.10.7) is equivalent to a gaussian prior model of the form (1.8.10) in a Bayesian formulation. Fig. 1.7.1 shows an example of a random image (a gaussian random field) drawn from this "prior," having covariance* $K_{\boldsymbol{x}} = \mathsf{R}^{-1}$*, where* $\mathsf{R} = C'C$*. This matrix is not quite invertible, so Fig. 1.7.1 used the pseudo-inverse of a circulant approximation to* R*. See Problem 1.27.*

s,res,penal2,nonquad

1.10.2 Nonquadratic (edge preserving) regularization

As discussed in §1.9, a quadratic roughness penalty causes blurring of object edges, because squaring the differences between neighboring pixels strongly encourages neighboring values to be similar. To try to preserve edges, we can replace the quadratic function[16] $\psi(z) = \frac{1}{2}|z|^2$ in (1.10.1) with a nonquadratic function ψ that rises less rapidly than the quadratic when the difference $|x_j - x_k|$ exceeds some user-selected threshold $\delta > 0$. One way to avoid over-regularizing the differences between neighboring pixels is to use the **Huber function** illustrated in Fig. 1.10.4:

e,res,penal2,huber

$$\psi(z) = \psi_\delta(z) \triangleq \begin{cases} \frac{1}{2}|z|^2, & |z| \le \delta \\ \delta|z| - \frac{1}{2}\delta^2, & |z| > \delta \end{cases} = \delta^2 \, \psi_1(z/\delta). \tag{1.10.9}$$

Often such **edge preserving** potential functions are quadratic near zero, but roughly linear far from zero. Large disparities between neighboring pixels, such as might arise near object edges, are penalized less by such a **potential function**[17] than by quadratic penalty functions. (See §1.10.3.)

It is possible to use different potential functions for different spatial locations. Therefore, most of the algorithms described in this *book* are derived for roughness penalty functions having the following general form:

e,res,Rx

$$\mathsf{R}(\boldsymbol{x}) = \sum_{k=1}^{K} \psi_k([C\boldsymbol{x}]_k), \tag{1.10.10}$$

where $[C\boldsymbol{x}]_k = \sum_{j=1}^{n_\mathrm{p}} c_{kj} x_j$. The matrix C is $K \times n_\mathrm{p}$ where $n_\mathrm{p} = MN$, and for the simple 2D case (1.10.1) with horizontal and vertical neighbors, we have $K = 2MN$. This form is sufficiently general to represent many, but not all, regularizers (and log priors) that have been described in the literature. §2.7 describes many other choices for ψ_k.

s,res,npls

1.10.3 Analysis of least-squares with nonquadratic regularization (s,res,npls)

Although analysis of the properties of penalized-likelihood estimates $\hat{\boldsymbol{x}}$ is more difficult when the regularizer is nonquadratic, we can still get some insight into the edge-preserving characteristics by deriving a recursive expression for $\hat{\boldsymbol{x}}$ as shown in this section.

We focus on penalized least-squares cost functions of the form

e,res,Kx,nqls

$$\Psi(\boldsymbol{x}) = \frac{1}{2}\|\boldsymbol{y} - A\boldsymbol{x}\|^2 + \beta\,\mathsf{R}(\boldsymbol{x}) \tag{1.10.11}$$

where the penalty function has the general form (1.10.10) but with $\psi_k = \psi$. In (1.10.11) we have dropped the noise variance σ^2 from (1.7.9) because it often is unknown and we can always absorb into β anyway. We will not consider circulant approximations here because nonquadratic penalty functions introduce local shift-variant effects.

For the analysis in this section, we assume the **potential function** ψ satisfies the following two conditions.

[16] I use the letter z as the argument of the potential function $\psi(z)$ because the regularizer encourages its argument to be close to zero.

[17] The term **potential function** for ψ is prevalent in the Bayesian restoration literature, but originates even earlier in physics literature on the energy of various configurations of spatial models. [wiki]

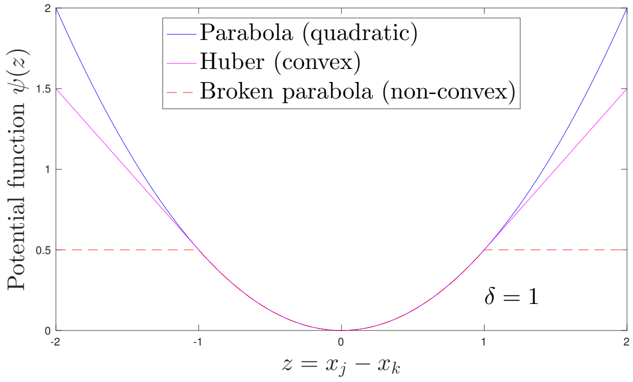


Figure 1.10.4: Illustration of some potential functions ψ used for regularization: the quadratic function, the Huber function, and the "broken-parabola" function (see §1.12.1). fig_broken

- ψ is everywhere differentiable with derivative $\dot{\psi}$.
- The following **weighting function** [88, p. 179] is defined (finite) and nonnegative for all $z \in \mathbb{R}$:

$$\omega_\psi(z) \triangleq \frac{\dot{\psi}(z)}{z}. \qquad (1.10.12)$$ e,res,npls,wpot

For later algorithm derivations and convergence analysis (*cf.* §12.4.4 and [89]) we will make stronger assumptions about ψ, but these two conditions suffice here. Fig. 1.10.5 illustrates the weighting functions ω_ψ for the potential functions shown in Fig. 1.10.4. (Fig. 2.7.1 illustrates many more from Table 2.1.) As detailed in §2.7, usually $\omega_\psi(z)$ is a decreasing function of $|z|$.

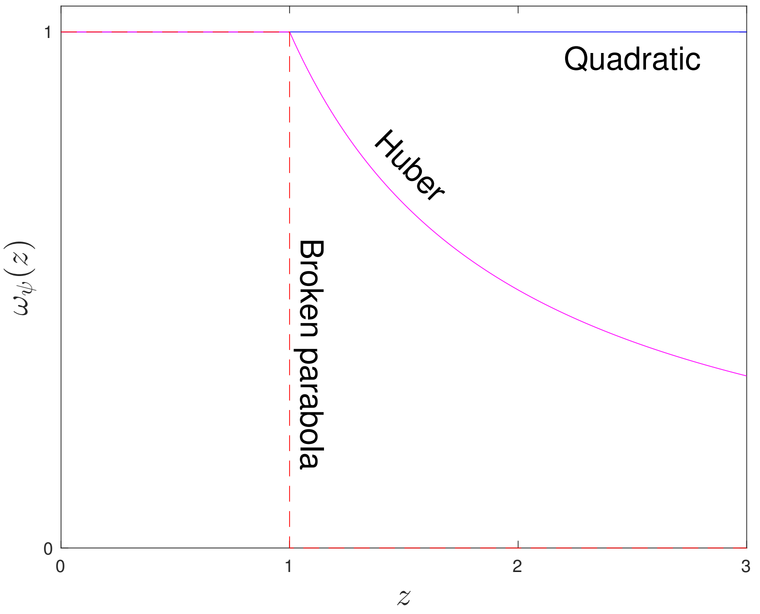


Figure 1.10.5: The potential weighting functions $\omega_\psi(z)$ for a quadratic potential, Huber potential, and a broken parabola potential, illustrated using $\delta = 1$. fig_res_wpot_huber

Typical edge-preserving potential functions have the property that they are nearly quadratic near zero, and nearly linear when the argument exceeds δ. The parameter δ controls the transition between smoothing and edge-preservation, so one must have in advance a rough idea of the anticipated differences between neighboring pixels that straddle region boundaries, or use trial and error to find δ. As elaborated below, the nonquadratic property will encourage most neighboring pixel values to be similar, but will also allow them to be different in image locations where there are sufficient discrepancies between neighbors, *i.e.*, near object edges.

To explore the properties further, first we examine the column gradient of the penalty function:

$$\nabla \mathsf{R}(\boldsymbol{x}) = \sum_{k=1}^{K} \nabla \psi(\boldsymbol{c}_k' \boldsymbol{x}) = \sum_{k=1}^{K} \boldsymbol{c}_k \, \dot{\psi}([\boldsymbol{C}\boldsymbol{x}]_k) = \sum_{k=1}^{K} \boldsymbol{c}_k \, \omega_\psi([\boldsymbol{C}\boldsymbol{x}]_k) [\boldsymbol{C}\boldsymbol{x}]_k = \boldsymbol{C}' \boldsymbol{D}(\boldsymbol{x}) \, \boldsymbol{C}\boldsymbol{x} \qquad (1.10.13)$$ e,res,R,cgrad

where $\boldsymbol{c}_k' = \boldsymbol{e}_k' \boldsymbol{C}$ denotes the kth row of $\boldsymbol{C}$ and we define the following $\boldsymbol{x}$-dependent, $K \times K$ diagonal weighting matrix:

$$\boldsymbol{D}(\boldsymbol{x}) \triangleq \mathsf{diag}\{\omega_\psi([\boldsymbol{C}\boldsymbol{x}]_k)\}. \qquad (1.10.14)$$ e,res,npls,Df

(The weighting function (1.10.12) was constructed to enable the final form (1.10.13).) Thus the gradient of the cost function (1.10.11) is

e,res,npls,cgcost

$$\nabla\Psi(\boldsymbol{x}) = -\boldsymbol{A}'\,(\boldsymbol{y}-\boldsymbol{A}\boldsymbol{x}) + \beta\boldsymbol{C}'\,\boldsymbol{D}(\boldsymbol{x})\,\boldsymbol{C}\boldsymbol{x}, \tag{1.10.15}$$

which equated to zero yields

$$\boldsymbol{A}'\boldsymbol{y} = \boldsymbol{A}'\boldsymbol{A}\,\hat{\boldsymbol{x}} + \beta\boldsymbol{C}'\,\boldsymbol{D}(\hat{\boldsymbol{x}})\,\boldsymbol{C}\,\hat{\boldsymbol{x}}\,.$$

Rearranging yields the following recursive expression for the estimator $\hat{\boldsymbol{x}}$:

e,res,npls,xh

$$\hat{\boldsymbol{x}} = \left[\boldsymbol{A}'\boldsymbol{A} + \beta\boldsymbol{C}'\,\boldsymbol{D}(\hat{\boldsymbol{x}})\,\boldsymbol{C}\right]^{-1}\boldsymbol{A}'\boldsymbol{y}. \tag{1.10.16}$$

This is a somewhat remarkable expression, but it is not quite in closed form because $\hat{\boldsymbol{x}}$ is present on the right-hand side. Nevertheless, it provides insight into the properties of nonquadratic penalty functions. (A closed form is known for the 1D case with ideal step-function data for the Huber potential function [90].)

First reconsider the quadratic case where $\psi(z) = z^2/2$. Then $\omega_\psi(z) = 1$ so $\boldsymbol{D}(\hat{\boldsymbol{x}}) = \boldsymbol{I}$ and we are back to the usual quadratic penalized least-squares estimator (1.8.9) with its edge-blurring properties.

For more insight into (1.10.16), consider again the 1D problem described in §1.8.1. If we knew where the edges were, we would use a *spatially weighted* penalty function [91]

e,res,npls,R,ideal

$$\mathsf{R}(\boldsymbol{x}) = \sum_{j=2}^{N} w_j\,\psi(x_j - x_{j-1}) \tag{1.10.17}$$

where we would set $w_j = 0$ between pixels straddling an edge, and $w_j = 1$ for pixels in relatively uniform regions. This would provide smoothing and hence noise reduction in uniform regions without causing regularization-induced blurring across edges. In the quadratic case where $\psi(z) = z^2/2$, this "oracle weighted" regularizer would lead to the solution

$$\hat{\boldsymbol{x}} = \left[\boldsymbol{A}'\boldsymbol{A} + \beta\boldsymbol{C}'\,\mathsf{diag}\{w_j\}\,\boldsymbol{C}\right]^{-1}\boldsymbol{A}'\boldsymbol{y},$$

which would provide ideal edge-preserving restoration. However, in practice we (usually) do not know where the edges are in advance, so we must "let the algorithm find them." Comparing this expression to (1.10.16), we see that the "only" difference is that in (1.10.16) the weights ω_ψ *depend on the estimate* $\hat{\boldsymbol{x}}$.

For example, consider the Huber **potential function** given in Table 2.1 and its corresponding **weighting function** shown in Fig. 1.10.5. When the difference $[\boldsymbol{C}\,\hat{\boldsymbol{x}}]_k$ between neighboring pixels exceeds δ, the corresponding weight is reduced from unity by ω_ψ, thereby approximating the effect described in (1.10.17). In other words, instead of needing to know the edge locations in advance, a nonquadratic penalty function can provide *estimate-based* weighting. (See Example 1.11.3 and Fig. 1.11.2 for an example.) In the **denoising** case where $\boldsymbol{A} = \boldsymbol{I}$, one can think of $\hat{\boldsymbol{x}}$ as an iterative form of adaptive smoothing.

There are explicitly adaptive methods for image restoration where one processes the image $\boldsymbol{y}$ to attempt to locate edge regions, *e.g.*, by using local statistics [92]–[97]. Such methods are applicable to image restoration problems where $\boldsymbol{y}$ is already an image, but not to other inverse problems such as tomography where $\boldsymbol{y}$ is not an image.

s,res,alg.tex

1.11 Minimization algorithms (s,res,alg.tex)

Typically closed-form solutions are unavailable for the minimizer $\hat{\boldsymbol{x}}$ of the cost functions (1.8.1) of interest in inverse problems, so finding $\hat{\boldsymbol{x}}$ requires iterative optimization algorithms. Chapters 11 and 12 describe such algorithms in detail; here we preview just two.

An **iterative algorithm** is a procedure that starts with an initial guess $\boldsymbol{x}^{(0)}$ for $\hat{\boldsymbol{x}}$, and then recursively generates a sequence[18] $\boldsymbol{x}^{(1)}, \boldsymbol{x}^{(2)}, \ldots$, also denoted $\{\boldsymbol{x}^{(n)}\}$. Ideally, the iterates $\{\boldsymbol{x}^{(n)}\}$ should rapidly approach the minimizer $\hat{\boldsymbol{x}}$.

1.11.1 Gradient-based algorithms

Most algorithms involve the gradient of the cost function $\Psi(\boldsymbol{x})$, as described in detail in Chapter 11, Many of the algorithms reduce to the following form:

e,res,alg

$$\boldsymbol{x}^{(n+1)} = \boldsymbol{x}^{(n)} - \alpha_n \boldsymbol{D}\,\nabla\Psi(\boldsymbol{x}^{(n)}), \tag{1.11.1}$$

where $\boldsymbol{D}$ is some diagonal **preconditioning matrix**, and α_n is a **step size** that affects the convergence rate of the sequence $\{\boldsymbol{x}^{(n)}\}$. Such an algorithm will be of limited use unless it converges to the solution $\hat{\boldsymbol{x}}$. To preview the convergence analyses described in later chapters, consider the case where $\Psi(\boldsymbol{x})$ is quadratic, such as in (1.7.5). Such functions can also be written in the following two forms:

$$\Psi(\boldsymbol{x}) = c - \boldsymbol{b}'\boldsymbol{x} + \frac{1}{2}\boldsymbol{x}'\boldsymbol{H}\boldsymbol{x} = \Psi(\hat{\boldsymbol{x}}) + \nabla\,\Psi(\hat{\boldsymbol{x}})\,(\boldsymbol{x}-\hat{\boldsymbol{x}}) + \frac{1}{2}(\boldsymbol{x}-\hat{\boldsymbol{x}})'\boldsymbol{H}\,(\boldsymbol{x}-\hat{\boldsymbol{x}})\,,$$

[18] Throughout this *book*, all superscripts (such as (n) and $(n+1)$) on vectors such as $\boldsymbol{x}$ denote iteration indices.

where $\boldsymbol{H} \triangleq \nabla^2 \Psi(\boldsymbol{x})$ is the **Hessian** of the cost function Ψ and the elements of $\boldsymbol{H}$ are given by $H_{jk} = \frac{\partial^2}{\partial x_j \partial x_k} \Psi(\boldsymbol{x})$. (For a quadratic cost function this Hessian is independent of $\boldsymbol{x}$.) Recalling that $\nabla \Psi(\hat{\boldsymbol{x}}) = \mathbf{0}$ in the absence of constraints, we have

$$\nabla \Psi(\boldsymbol{x}^{(n)}) = \boldsymbol{H} \left(\boldsymbol{x}^{(n)} - \hat{\boldsymbol{x}}\right).$$

Subtracting $\hat{\boldsymbol{x}}$ from both sides of (1.11.1) and substituting in the above yields

e,res,alg,xn-xh

$$\boldsymbol{x}^{(n+1)} - \hat{\boldsymbol{x}} = \boldsymbol{x}^{(n)} - \hat{\boldsymbol{x}} - \alpha_n \boldsymbol{D} \boldsymbol{H} \left(\boldsymbol{x}^{(n)} - \hat{\boldsymbol{x}}\right) = (\boldsymbol{I} - \alpha_n \boldsymbol{D} \boldsymbol{H})(\boldsymbol{x}^{(n)} - \hat{\boldsymbol{x}}). \tag{1.11.2}$$

So the distance $\boldsymbol{x}^{(n)} - \hat{\boldsymbol{x}}$ to the solution should decrease each iteration if the matrix $\boldsymbol{I} - \alpha_n \boldsymbol{D} \boldsymbol{H}$ is "small" in some appropriate sense.

For further insight, consider the penalized least-squares cost function (1.8.8) and suppose that $\boldsymbol{A}$ and R are circulant, $\boldsymbol{D} = \boldsymbol{I}$, and $\boldsymbol{K}_{\boldsymbol{\varepsilon}} = \sigma^2 \boldsymbol{I}$. Then the Hessian is

$$\boldsymbol{H} = \frac{1}{\sigma^2} \boldsymbol{A}' \boldsymbol{A} + \beta \mathsf{R} = \boldsymbol{Q}^{-1} \left[\frac{1}{\sigma^2} \boldsymbol{\Gamma}' \boldsymbol{\Gamma} + \beta \boldsymbol{\mathsf{\Pi}} \right] \boldsymbol{Q},$$

where $\boldsymbol{Q}$ is a DFT matrix and $\boldsymbol{\Gamma}$ and $\boldsymbol{\mathsf{\Pi}}$ were defined in §1.8.3. Let $\boldsymbol{E}^{(n)} \triangleq \boldsymbol{Q} \left(\boldsymbol{x}^{(n)} - \hat{\boldsymbol{x}}\right)$ denote the DFT coefficients of the error vector at the nth iteration. Then from the recursion (1.11.2) we have

$$\boldsymbol{E}^{(n+1)} = \left(\boldsymbol{I} - \alpha_n \left(\frac{1}{\sigma^2} \boldsymbol{\Gamma}' \boldsymbol{\Gamma} + \beta \boldsymbol{\mathsf{\Pi}} \right) \right) \boldsymbol{E}^{(n)}.$$

Therefore, all error frequency components will decrease at the nth iteration if and only if one chooses α_n such that

$$\left| 1 - \alpha_n \left(\frac{|B_k|^2}{\sigma^2} + \beta \mathsf{R}_k \right) \right| < 1, \qquad \forall k.$$

Of course, in the circulant case with a quadratic cost function, there is no need to use an iterative algorithm in the first place because a direct solution of the form (1.8.15) is available. But even in nonquadratic, non-circulant problems, similar convergence conditions arise, as discussed in more detail in Chapters 11 and 14.

s,res,npls,alg

1.11.2 Huber's iteration (s,res,npls,alg)

The recursive form of the "solution" (1.10.16) is suggestive of the following fixed-point iteration:

$$\begin{aligned}
\boldsymbol{x}^{(n+1)} &= \left[\boldsymbol{A}'\boldsymbol{A} + \beta \boldsymbol{C}' \boldsymbol{D}(\boldsymbol{x}^{(n)}) \boldsymbol{C}\right]^{-1} \boldsymbol{A}'\boldsymbol{y} \\
&= \boldsymbol{x}^{(n)} - \left[\boldsymbol{A}'\boldsymbol{A} + \beta \boldsymbol{C}' \boldsymbol{D}(\boldsymbol{x}^{(n)}) \boldsymbol{C}\right]^{-1} \left(\left[\boldsymbol{A}'\boldsymbol{A} + \beta \boldsymbol{C}' \boldsymbol{D}(\boldsymbol{x}^{(n)}) \boldsymbol{C}\right] \boldsymbol{x}^{(n)} - \boldsymbol{A}'\boldsymbol{y} \right) \\
&= \boldsymbol{x}^{(n)} - \left[\boldsymbol{A}'\boldsymbol{A} + \beta \boldsymbol{C}' \boldsymbol{D}(\boldsymbol{x}^{(n)}) \boldsymbol{C}\right]^{-1} \nabla \Psi(\boldsymbol{x}^{(n)}),
\end{aligned} \tag{1.11.3}$$

using (1.10.15). This is a form of **preconditioned gradient descent**. Often fixed-point iterations do not converge, but this algorithm, derived by Huber [88, p. 182], decreases $\Psi(\boldsymbol{x})$ monotonically each iteration, and converges to the minimizer of $\Psi(\boldsymbol{x})$ under mild conditions; see §12.4.4.4 and Chapter 13.

However, this algorithm is somewhat impractical for imaging problems due to the matrix inverse. Chapter 12 describes several practical alternatives. A particularly simple choice is the diagonally-preconditioned gradient descent method (12.5.13), which has the form

$$\boldsymbol{x}^{(n+1)} = \boldsymbol{x}^{(n)} - \operatorname{diag}\left\{ \frac{1}{d_j^{(n)}} \right\} \nabla \Psi(\boldsymbol{x}^{(n)}) \tag{1.11.4}$$

$$d_j^{(n)} = \sum_{i=1}^{n_{\mathrm{d}}} |a_{ij}| \, |a|_i + \beta \sum_{k=1}^{K} |c_{kj}| \, |c|_k \, \omega_\psi([\boldsymbol{C}\boldsymbol{x}^{(n)}]_k), \tag{1.11.5}$$

where

$$|a|_i \triangleq \sum_{j=1}^{n_{\mathrm{p}}} |a_{ij}|, \qquad |c|_k \triangleq \sum_{j=1}^{n_{\mathrm{p}}} |c_{kj}|.$$

This iteration was used for the following example and for Fig. 1.11.1.

s,res,ex1

1.11.3 Restoration example (s,res,ex1)

Fig. 1.11.1 illustrates the methods summarized in this chapter. The normalized root mean-squared (NRMS) errors of each method are shown along the vertical axis of each image, where

$$\text{NRMS} \triangleq \frac{\sqrt{\sum_{j=1}^{n_{\mathrm{p}}} |\hat{x}_j - x_j|^2}}{\sqrt{\sum_{j=1}^{n_{\mathrm{p}}} |x_j|^2}}.$$

This figure of merit leaves much to be desired as a measure of image quality, but it serves as a starting point. Later chapters discuss more interesting figures of merit such as resolution (Chapter 22), noise (Chapter 23), and signal detection (Chapter 21).

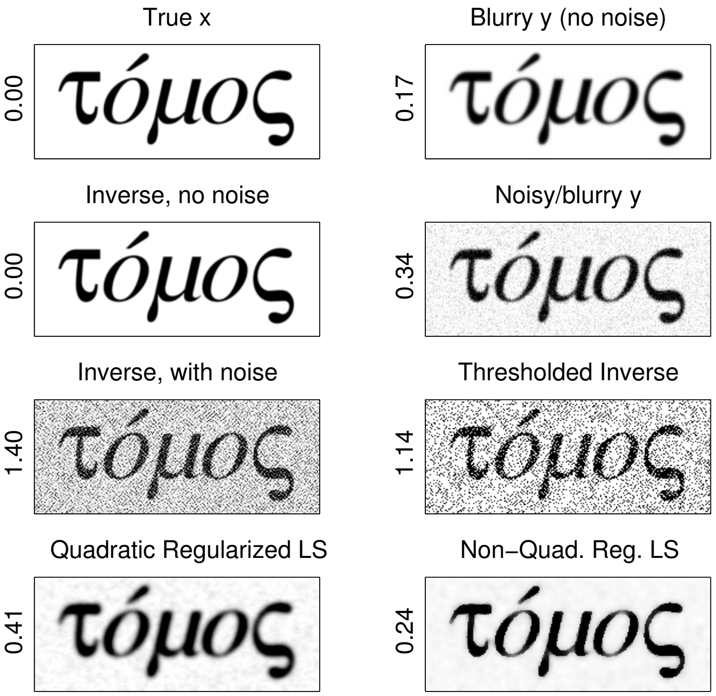


Figure 1.11.1: Illustration of 2D image restoration by: unregularized methods, quadratically regularized least-squares methods, and edge-preserving nonquadratically regularized least-squares methods. Each vertical axis shows the corresponding NRMS error.

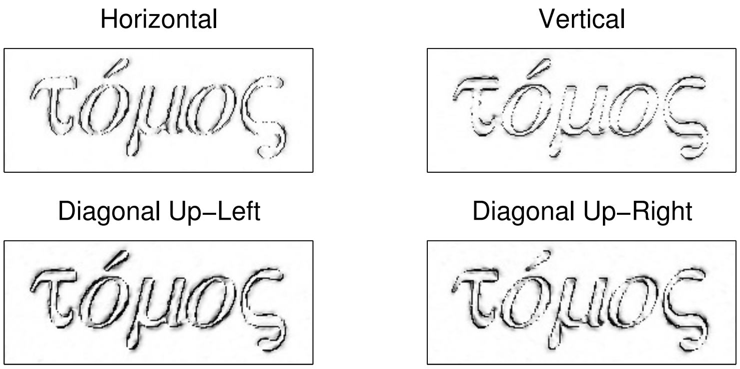
fig_tomos

The figure includes the following images.

- The true object $\boldsymbol{x}_{\text{true}}$, a grayscale image of the Greek word $\tau\acute{o}\mu o\sigma$ ("a slice", the root of **tomography**) where white=100 and black=0.
- Noiseless blurry image $\bar{\boldsymbol{y}} = \boldsymbol{A}\boldsymbol{x}$ where $\boldsymbol{A}$ corresponds to shift-invariant blur having the separable PSF $b[m,n] = b[n]\, b[m]$, where $b[n] = \left[\frac{1}{9}, \frac{1}{9}, \underline{\frac{5}{9}}, \frac{1}{9}, \frac{1}{9}\right]$.
- Noisy blurry image $\boldsymbol{y} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{\varepsilon}$ where the noise is zero-mean additive white gaussian distributed with $\sigma = 10$.
- The ML (inverse filter) reconstruction $\hat{\boldsymbol{x}}_{\text{ML}}$ defined in (1.6.9), computed via a circulant deconvolution approximation, both from the noiseless data $\bar{\boldsymbol{y}}$ and the noisy data $\boldsymbol{y}$. For noiseless data, the deconvolution method works fine, but even for this moderately noisy data there is excessive noise amplification.
- In this case the image is (nearly) binary, so we can improve on $\hat{\boldsymbol{x}}_{\text{ML}}$ by rounding each pixel value in $\hat{\boldsymbol{x}}_{\text{ML}}$ to 0 or 100, whichever is nearer. (This is the ML classifier ignoring pixel correlations introduced by the deconvolution.) However, this only reduces the NRMS error from 141% to 114%.
- A quadratically penalized least-squares estimator computed using the first-order roughness penalty (1.10.1) in the penalized-likelihood cost function (1.8.1) which reduces to (1.8.8) because the noise is gaussian. The quadratic regularization reduces noise relative to $\boldsymbol{y}$, but the edges are so seriously blurred that the overall NRMS error, which accounts for both noise and resolution effects, actually increases relative to $\boldsymbol{y}$.
- Finally, a nonquadratically penalized least-squares estimate with hyperbolic potential function and a 2nd-order neighborhood (including nearest diagonal neighbors). This approach gives the lowest NRMS of all methods shown for the noisy data. This example is ideal for such edge-preserving regularization because the true object is piece-wise smooth, indeed it is piece-wise constant.

Fig. 1.11.2 shows the weighting terms $\omega_\psi([\boldsymbol{C}\,\hat{\boldsymbol{x}}]_k)$ for the nonquadratic restoration example shown in Fig. 1.11.1. The weights are greatly reduced towards zero (black) from unity (white) near object edges, thereby helping preserve the edges in the restored image $\hat{\boldsymbol{x}}$.

IRT *See* `restore_example.m`.



fig_tomos_wpot

Figure 1.11.2: Weighting terms $\omega_\psi([\boldsymbol{C}\,\hat{\boldsymbol{x}}]_k)$ associated with Fig. 1.11.1.

[RQ]

1.12 Other formulations

s,res,line

1.12.1 Bayesian line-site models (s,res,line)

A nonquadratic but **convex potential function** can only *partially* preserve edges because large disparities between neighboring pixels are still discouraged, just less vigorously than the quadratic penalty function would. To penalize edges even less, one could use a **non-convex potential function** such as a **broken-parabola** function:

e,res,broken

$$\psi(z) = \begin{cases} |z|^2/2, & |z| < \delta \\ \delta^2/2, & |z| \geq \delta. \end{cases} \tag{1.12.1}$$

This potential function, illustrated in Fig. 1.10.4, assigns the same cost ($\delta^2/2$) to *any* value of the difference between two neighboring pixels exceeding δ.

Such potential functions are non-convex, which complicates the problem of finding the minimizer of the cost function $\Psi(\boldsymbol{x})$. Nevertheless, for piece-wise constant images, impressive restoration results have been reported.

Such methods originated in the Bayesian image restoration literature, but can be described equally well in the language of penalized-likelihood estimation. For neighboring pixels j and k, let l_{jk} denote a "**line site**" variable that should indicate the presence of an object edge "between" the jth and kth pixel. Typically we would like to have $l_{jk} = 1$ if there is an object intensity edge between pixels j and k, and to have $l_{jk} = 0$ if pixels j and k are both within an object region having uniform values. Intermediate values of l_{jk} can also be used if desired. Of course in practice we do not know which pixels are in the same region and which pairs straddle object edges, so we must estimate the l_{jk} values along with the object $\boldsymbol{x}$.

If we *did* know good l_{jk} values, then in the same spirit as (1.10.17) we would want to use a roughness penalty function of the following form to avoid smoothing across edges:

e,res,line,Rxl

$$\mathsf{R}(\boldsymbol{x}, \boldsymbol{l}) = \sum_{j=1}^{n_{\mathrm{p}}} \sum_{k \in \mathcal{N}_j} (1 - l_{jk}) \frac{1}{2} |x_j - x_k|^2 , \tag{1.12.2}$$

where $\mathcal{N}_j$ denotes (half of) the set of neighbors of the jth pixel, as defined in §1.10, and where $\boldsymbol{l} = \{l_{jk}\}$.

This formulation increases the number of unknowns from the n_{p} elements of $\boldsymbol{x}$ to at least $(1+2)n_{\mathrm{p}}$ unknowns if we use l_{jk} parameters corresponding to horizontal and vertical neighbors. To provide a useful solution, we must include some type of penalty function for the l_{jk} values as well, to discourage unlikely configurations of edges. Let $U(\boldsymbol{l})$ denote this penalty function. Then the overall penalized-likelihood estimate for such an approach is given by:

$$\begin{aligned} (\hat{\boldsymbol{x}}, \hat{\boldsymbol{l}}) &= \arg\min_{\boldsymbol{x}, \boldsymbol{l}} \Psi(\boldsymbol{x}, \boldsymbol{l}) \\ \Psi(\boldsymbol{x}, \boldsymbol{l}) &= -\log \mathsf{p}(\boldsymbol{y} \,|\, \boldsymbol{x}) + \beta_1 \, \mathsf{R}(\boldsymbol{x}, \boldsymbol{l}) + \beta_2 U(\boldsymbol{l}), \end{aligned} \tag{1.12.3}$$

where each l_{jk} is constrained to the interval $[0, 1]$. One can show that as $\beta_2 \to 0$, $\hat{l}_{jk} \to 1$, because $\hat{l}_{jk} = 1$ minimizes (1.12.2), and $\hat{\boldsymbol{x}}$ degenerates to the ML estimate. So a nonzero β_2 and an appropriate $U(\boldsymbol{l})$ are essential for this approach.

The simplest approach is to choose $U(\boldsymbol{l})$ to count the number of edge sites:

e,res,U(l),noninteract

$$U(\boldsymbol{l}) = \sum_{j,k} l_{jk}. \tag{1.12.4}$$

This penalty function discourages the formation of "too many" edges, but is indifferent to the shape of region boundaries. This is called a **non-interacting** line-site model. One can show [98]–[101] that this non-interacting model simplifies to ordinary penalized-likelihood estimation with the broken-parabola (1.12.1) as the potential function (Problem 1.13). One can show that replacing $\frac{1}{2}|x_j - x_k|^2$ with $|x_j - x_k|$ in (1.12.2) for a non-interacting model simplifies to ordinary PL estimation with a truncated absolute difference potential function [102] of the form $\psi(z) = \min(|z|, \beta_2/\beta_1)$.

To somewhat encourage object boundary continuity, line-site interactions are needed in the design of $U(\boldsymbol{l})$. Most papers *e.g.*, [103]–[106], have used small **cliques** that have an inherently local effect and thus only partially encourage boundary continuity. There are also similar variational formulations, *e.g.*, [107].

In the context of blurred image restoration, comparatively large line-site neighborhood sizes that match the size of the PSF of the imaging system have been proposed [108], [109]. (How to apply that principle in tomography problems is unclear because each tomographic measurement is influenced by long strips traversing the entire object.) None of these line-site models address *global* connectivity or continuity of object boundaries, and thus are inherently local.

One of the few previous methods to capture global properties is a region-based Bayesian prior that has been applied successfully in tomography [110], [111]. That method uses discrete region identifiers (motivated by image segmentation problems) and assigns costs that prohibit disconnected regions, encourage regularly shaped regions, and discourage having too many regions. (The number of regions need not be specified *a priori*.) Some of these costs involve the entire image and are therefore global. The Bayesian formalism permits the exploration of estimate uncertainty, but using discrete region labels is challenging for computing point estimates.

An alternative method to encourage global boundary regularity is to use level sets to model object boundaries [112]–[114]. Such approaches are boundary based rather than region based; boundaries are continuous-valued, so simple gradient-descent methods are available for computing point estimates (at local minima of the cost function).

This book focuses on penalized-likelihood problems of the form (1.8.1) rather than on the line-site form (1.12.3). Nevertheless, many of the concepts would also apply to line-site formulations. When restoring multiple images simultaneously with common boundaries, line-site methods may be useful [115]. See §2.8.

s,res,maxent

1.12.2 Maximum entropy restoration (s,res,maxent)

An alternative to maximum-likelihood and penalized-likelihood methods are the **maximum entropy** methods for image restoration [116]–[129]. These methods use a version of **entropy** [wiki] to quantify image smoothness or irregularity. (A uniform image, when treated as a uniform distribution, has high entropy, whereas an image that is a single Kronecker impulse has zero entropy in this framework.) This approach has been applied in many application domains, including spectral estimation [130] and tomographic reconstruction, *e.g.*, [44], [131]–[135].

Some maximum entropy formulations ignore measurement noise, *e.g.*, [136], as follows:

$$\hat{\boldsymbol{x}} = \operatorname*{arg\,max}_{\boldsymbol{x} \succeq \boldsymbol{0}} - \sum_{j=1}^{n_{\mathrm{p}}} x_j \log x_j \text{ such that } \boldsymbol{y} = \boldsymbol{A}\boldsymbol{x}.$$

Other formulations allow for some noise, *e.g.*,

$$\hat{\boldsymbol{x}} = \operatorname*{arg\,max}_{\boldsymbol{x} \succeq \boldsymbol{0}} - \sum_{j=1}^{n_{\mathrm{p}}} x_j \log x_j \text{ such that } \|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}\|^2 \leq L\sigma^2,$$

where L is a parameter that one must select. By the theory of constrained optimization [47, p. 188], this latter approach is equivalent to a form of penalized least-squares estimation:

$$\hat{\boldsymbol{x}} = \operatorname*{arg\,min}_{\boldsymbol{x} \succeq \boldsymbol{0}} \frac{1}{2} \|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}\|^2 + \beta \sum_{j=1}^{n_{\mathrm{p}}} x_j \log x_j,$$

for some value of β that depends on L above. In this setting, the (negative) entropy penalty function $\mathsf{R}(\boldsymbol{x}) = \sum_{j=1}^{n_{\mathrm{p}}} x_j \log x_j$ is just one of many possible choices, as is not in any sense canonical. Yet another option (particularly for the under-determined case) is to choose, among all the LS solutions, the one having maximum entropy. Lyon et al. attempt this by minimizing $\mathsf{R}(\boldsymbol{x})$ subject to $\boldsymbol{x}'\boldsymbol{1} = 1$, $\boldsymbol{x} \succeq \boldsymbol{0}$, and $-\boldsymbol{A}'(\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}) = \boldsymbol{0}$. The latter expression characterizes the family of *unconstrained* LS solutions, but not the set of nonnegativity-constrained LS solutions, so formulating maxent rigorously remains an *open problem*.

Although there are philosophical arguments that might be made in favor of maximum entropy in the absence of measurement noise, those arguments seem weaker in the presence of noise. The maximum entropy method is popular in astronomical image restoration because it favors "nearly black" objects [137], [138]. It is appropriately less popular in the tomographic imaging field.

s,res,sparse 1.12.3 Sparsity models (s,res,sparse)

s,res,sparse,synth 1.12.3.1 Synthesis formulations

In some applications the image itself is expected to be **sparse**, *e.g.*, when imaging some star fields [139] or in angiography [140]. In other applications, one might be willing to assume that the object is **sparse** with respect to some basis $\boldsymbol{B}$. In such applications, we can "synthesize" $\boldsymbol{x}$ in terms of a small number of basis vectors by writing

e,res,sparse,synth

$$\boldsymbol{x} = \boldsymbol{B}\boldsymbol{z}, \tag{1.12.5}$$

where $\boldsymbol{B}$ denotes a $n_{\mathrm{p}} \times K$ matrix whose columns are basis functions and $\boldsymbol{z}$ is an unknown vector of coefficients. In some cases $\boldsymbol{B}$ is **overcomplete**, meaning $K > n_{\mathrm{p}}$. (In those cases, $\boldsymbol{B}$ is no longer a basis and often one uses generalizations called **frames** [141].) Sometimes $\boldsymbol{B}$ is selected to be a standard basis such as wavelets or the **discrete cosine transform** (**DCT**). In other cases, one learns $\boldsymbol{B}$ from the data [142]–[145]. In methods based on sparsity assumptions, one expects only a small subset of the elements of $\boldsymbol{z}$ to be nonzero. Let $\|\boldsymbol{z}\|_0$ denote[19] the number of nonzero elements of $\boldsymbol{z}$, *i.e.*,

e,norm0

$$\|\boldsymbol{z}\|_0 \triangleq \sum_{k=1}^{K} \mathbb{I}_{\{z_k \neq 0\}}. \tag{1.12.6}$$

For such models, reasonable approaches to image recovery include the following.

- Finding the image that best fits the data subject to a sparsity constraint:

  $$\hat{\boldsymbol{z}} = \arg\min_{\boldsymbol{z}} \|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{B}\boldsymbol{z}\| \text{ s.t. } \|\boldsymbol{z}\|_0 \leq L \in \mathbb{N},$$

  where $L < K$. An **iterative hard thresholding algorithm** for this problem is derived in [146] using the **optimization transfer** methods of Chapter 12. However, rarely is the degree of sparsity L known.
- Finding the sparsest set of coefficients for which the corresponding image fits the data exactly [140], [147]:

  $$\hat{\boldsymbol{z}} = \arg\min_{\boldsymbol{z}} \|\boldsymbol{z}\|_0 \text{ s.t. } \boldsymbol{y} = \boldsymbol{A}\boldsymbol{B}\boldsymbol{z}.$$

  (Presumably this is most useful only in the unlikely situation that the data is noiseless!)
- Finding the sparsest set of coefficients for which the corresponding image fits the data to within some constraint:

  e,res,sparse,l0,eps

  $$\hat{\boldsymbol{z}} = \arg\min_{\boldsymbol{z}} \|\boldsymbol{z}\|_0 \text{ s.t. } \|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{B}\boldsymbol{z}\| \leq \varepsilon. \tag{1.12.7}$$

  This may be particularly well suited to highly under-determined problems [148].
- Finding the image that minimizes a cost function that includes a data fidelity term and sparsity regularizer:

  e,res,sparse,reg0

  $$\hat{\boldsymbol{z}} = \arg\min_{\boldsymbol{z}} \frac{1}{2} \|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{B}\boldsymbol{z}\|^2 + \beta \|\boldsymbol{z}\|_0 . \tag{1.12.8}$$

  Often this regularized approach is equivalent, for suitable β and ϵ. to the constrained approach (1.12.7), but not always (in nonconvex problems [149], [150]). The units of ϵ are more intuitive than those of β so (1.12.7) may simplify parameter selection. However, (1.12.7) is a constrained optimization problem so it may be more challenging than (1.12.8).
- Including both a sparsity regularizer and another regularizer:

  e,res,sparse,synth,R

  $$\hat{\boldsymbol{z}} = \arg\min_{\boldsymbol{z}} \frac{1}{2} \|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{B}\boldsymbol{z}\|^2 + \beta_1 \|\boldsymbol{z}\|_0 + \beta_2 \, \mathsf{R}(\boldsymbol{B}\boldsymbol{z}) . \tag{1.12.9}$$

  All of the above estimates are in the **synthesis form** because after finding the estimate $\hat{\boldsymbol{z}}$ of the coefficients we synthesize the final image by $\hat{\boldsymbol{x}} = \boldsymbol{B}\hat{\boldsymbol{z}}$ per (1.12.5).

1.12.3.2 Analysis formulations

An alternative to the synthesis formulations is to use an **analysis form** such as the following regularized cost function:

e,res,sparse,anal,R

$$\hat{\boldsymbol{x}} = \arg\min_{\boldsymbol{x}} \frac{1}{2} \|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}\|^2 + \beta_1 \|\boldsymbol{B}'\boldsymbol{x}\|_0 + \beta_2 \, \mathsf{R}(\boldsymbol{x}), \tag{1.12.10}$$

where now the columns of $\boldsymbol{B}$ are used to "**analyze**" the image $\boldsymbol{x}$ and penalize the number of nonzero coefficients. One can also define analysis formulations of the other variations in §1.12.3.1.

[19] This norm notation is a slight abuse, because $\|\boldsymbol{x}\|_p \triangleq \left(\sum_j |x_j|^p\right)^{1/p}$ is a norm when $p \geq 1$ but not when $p < 1$. Nevertheless, when $p < 1$ the summation is still well defined, and one can show that $\|\boldsymbol{x}\|_p^p$ converges to $\|\boldsymbol{x}\|_0$ as $p \to 0$. Technically (1.12.6) is called **counting measure**, not a norm.

When $\boldsymbol{B}$ is invertible, the analysis form (1.12.10) and synthesis form (1.12.9) are equivalent, but otherwise they differ and it is unknown which approach is best [151].

If $\boldsymbol{B}$ corresponds to the undecimated Haar analysis wavelets at the finest scale, which (in 1D) use the filter $\delta[n] - \delta[n-1]$, then this latter form is closely related to the roughness regularizer (1.8.3).

All of these approaches have been investigated for imaging problems. The "norm" $\|\boldsymbol{z}\|_0$ is nonconvex and nondifferentiable, greatly complicating minimization. Often it is replaced by $\|\boldsymbol{z}\|_1$, which is convex. For example, often we replace the ℓ_0 regularized minimization problem (1.12.8) with:

e,res,sparse,reg1

$$\hat{\boldsymbol{z}} = \arg\min_{\boldsymbol{z}} \frac{1}{2} \|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{B}\boldsymbol{z}\|^2 + \beta \|\boldsymbol{z}\|_1 . \tag{1.12.11}$$

This problem is known as **least absolute shrinkage and selection operator** (**LASSO**) in the statistics literature [152]–[157]. This cost function also is non-differentiable at the origin, so conventional optimization algorithms for it can be quite slow. This is a very active research area with a rapidly growing literature. See [158]–[167].

s,res,super

1.12.4 Super-resolution problems (s,res,super)

This chapter has focused on restoring an image $\boldsymbol{x}$ from a single noisy, blurred image $\boldsymbol{y}$. In some applications one can acquire *multiple* images $\boldsymbol{y}_1, \ldots, \boldsymbol{y}_K$ each of which is related to $\boldsymbol{x}$ by a possibly different system model $\boldsymbol{A}_k$ and noise model:

$$\boldsymbol{y}_k = \boldsymbol{A}_k \boldsymbol{x} + \boldsymbol{\varepsilon}_k .$$

If the system models differ by subpixel spatial translations, then one can attempt to recover $\boldsymbol{x}$ at a finer spatial resolution than the measured images $\{\boldsymbol{y}_k\}$. This problem is known as **super-resolution**. A typical approach is based on a cost function of the form (for white gaussian noise):

$$\Psi(\boldsymbol{x}) = \sum_{k=1}^{K} \frac{1}{2} \|\boldsymbol{y}_k - \boldsymbol{A}_k \boldsymbol{x}\|^2 + \mathsf{R}(\boldsymbol{x}) .$$

In many cases, the system model $\boldsymbol{A}_k$ depends on motion parameters that one also must estimate. See this article on dual-lens smart phone cameras, references [168]–[184] and these surveys [185], [186]. For medical imaging applications, including some controversies, see [187]–[196].

s,res,summ

1.13 Summary (s,res,summ)

This chapter has used the image restoration application as a vehicle for introducing many of the concepts and notational conventions that are used throughout this *book* for image reconstruction problems. The principal concepts include

- the deficiencies of inverse filtering (deconvolution),
- matrix-vector representation of linear models for imaging systems,
- circulant approximations for linear shift-invariant problems,
- ML, MAP, and penalized-likelihood estimation criteria,
- and roughness penalty functions and their effects.

The field of image restoration remains an active area of research with a growing number of applications.

s,res,prob

1.14 Problems (s,res,prob)

p,res,blur,fbasis

Problem 1.1 *An image restoration method uses object model (1.3.6) with a 2D rectangular basis function:* $\beta_0(x, y) = \mathrm{rect}_2(x/\triangle_X, y/\triangle_Y)$. *The sensor is shift invariant with a rectangular blur:* $b(x, y) = \frac{1}{4\triangle_X\triangle_Y}\mathrm{rect}_2\left(\frac{x}{2\triangle_X}, \frac{y}{2\triangle_Y}\right)$. *Assuming zero end conditions, determine the values of elements* a_{ij} *of the system matrix* $\boldsymbol{A}$. *Assume both the spacing of the sensor elements and the spacing of the object basis functions are* $(\triangle_X, \triangle_Y)$.

Problem 1.2 *Consider a 1D signal restoration problem for a linear shift-invariant system with additive white gaussian noise and impulse response*

$$b[n] = \delta[n-2] + 8\,\delta[n] + \delta[n+2]\,.$$

Would the deconvolution method be a suitable restoration approach for this system? Explain why or why not.

p,circ,ht

Problem 1.3 *Derive the circular convolution formulas (1.4.11) and (1.4.12) for periodic boundary conditions.*

Problem 1.4 *Use the principles of §1.4.3.1 to find analytically the eigenvalues and eigenvectors of the following matrix*

$$\begin{bmatrix} 5 & 2 & 1 & 0 & 1 & 2 \\ 2 & 5 & 2 & 1 & 0 & 1 \\ 1 & 2 & 5 & 2 & 1 & 0 \\ 0 & 1 & 2 & 5 & 2 & 1 \\ 1 & 0 & 1 & 2 & 5 & 2 \\ 2 & 1 & 0 & 1 & 2 & 5 \end{bmatrix}.$$

Verify your eigenvalues using MATLAB*'s* `eig` *function. Your* eigenvectors *may differ from those of* MATLAB*; why?*

Problem 1.5 *Consider a 1D system with impulse response* $b[n] = 3\,\delta[n] + \delta[n-1] + \delta[n+1]$. *Using* MATLAB*'s* `eig` *command, compute and plot the "exact" eigenvalues of the Toeplitz representation (1.4.7) for* $N = 8$, $N = 16$, *and* $N = 128$. *Determine analytically the eigenvalues of the circulant approximation (1.4.13) for* $N = 1000$ *and superimpose on the preceding plot. For superposition, sort the eigenvalues from largest to smallest, and normalize the eigenvalue indices as* $(k-1)/N$. *Discuss.*

p,circ,vs,toep3b

Problem 1.6 *[198, Eqn. (2.7)] states that* $\lambda_{\min}(\boldsymbol{A}) \leq \lambda_{\min}(c(\boldsymbol{A})) \leq \lambda_{\max}(c(\boldsymbol{A})) \leq \lambda_{\max}(\boldsymbol{A})$, *where* $c(\boldsymbol{A})$ *is a certain circulant approximation to* $\boldsymbol{A}$. *These inequalities provide the following bound on the* ***condition number*** *(26.7.2):*

e,res,circ,cond

$$\kappa(\boldsymbol{A}'\boldsymbol{A}) = \frac{\lambda_{\max}(\boldsymbol{A}'\boldsymbol{A})}{\lambda_{\min}(\boldsymbol{A}'\boldsymbol{A})} \geq \frac{\lambda_{\max}(c(\boldsymbol{A}'\boldsymbol{A}))}{\lambda_{\min}(c(\boldsymbol{A}'\boldsymbol{A}))} = \kappa(c(\boldsymbol{A}'\boldsymbol{A})). \tag{1.14.1}$$

Evaluate these bounds for the preceding problem and compare to the eigenvalues of the Toeplitz matrices. See also [199], [200]. *(Need typed.)*

Problem 1.7 *Can you find* bounds *that are the other way around? Possible resources: [198], [201]–[204] and citations therein.* *(Solve?)*

p,res,mirror,a

Problem 1.8 *Determine* $\boldsymbol{A}$ *for the* ***mirror end conditions*** *discussed in §1.4.* *(Need typed.)*

p,res,mat2,rep

Problem 1.9 *Generalize the expression (1.4.20) or (1.4.21) to the case of* ***replicated end conditions****.* *(Solve?)*

p,res,non,circ

Problem 1.10 *If a matrix* $\boldsymbol{M}$ *is square and circulant, then computing* $\boldsymbol{Q}\boldsymbol{M}\boldsymbol{Q}^{-1}$ *will yield an exactly diagonal matrix, where* $\boldsymbol{Q}$ *is the DFT matrix defined in (1.4.28). Consider the following four representations of a system matrix* $\boldsymbol{A}$*: (1.4.7), (1.4.9), (1.4.10), and (1.4.13). For each representation, using* MATLAB *to compute* $\boldsymbol{D} = \boldsymbol{Q}\boldsymbol{A}'\boldsymbol{A}\boldsymbol{Q}^{-1}$ *for the impulse response* $b[n] = \delta[n-1] + 2\,\delta[n] + \delta[n+1]$ *and for* $N = 64$. *(Hint: you can create each of the* $\boldsymbol{A}$ *matrices needed in one or two lines of* MATLAB *using* `convmtx`*.) Display for yourself the* $\boldsymbol{D}$ *matrices to visualize how close to diagonal they are. Compute the fractional off-diagonal "energy" as follows:*

$$\frac{\sqrt{\sum_{k \neq j} |d_{kj}|^2}}{\sqrt{\sum_{k,j} |d_{kj}|^2}}.$$

Compare the four models using this quantitative measure of "non-circulant-ness."

p,res,poisson,ml

Problem 1.11 *Consider the statistical model* $y_i \sim \mathsf{Poisson}\{a_i x\}$ *where* x *is an unknown nonnegative scalar and* $a_i = i^2$, $i = 1, \ldots, 5$. *Find expressions for the log-likelihood and the ML estimate of* x *given* $y_1, \ldots, y_5$.

p,res,orth,shrink

Problem 1.12 *For any* $n_p \times n_p$ *unitary matrix* $\boldsymbol{U}$, *consider the regularized LS cost function*

$$\Psi(\boldsymbol{z}) = \frac{1}{2}\|\boldsymbol{y} - \boldsymbol{U}\boldsymbol{z}\|^2 + \beta\sum_{j=1}^{n_p}\psi(z_j)\,.$$

- *Defining* $\boldsymbol{b} = \boldsymbol{U}'\boldsymbol{y}$, *and using the fact that*

$$\|\boldsymbol{y} - \boldsymbol{U}\boldsymbol{z}\|^2 = \|\boldsymbol{U}'\boldsymbol{y} - \boldsymbol{z}\|^2 = \|\boldsymbol{b} - \boldsymbol{z}\|^2 = \sum_{j=1}^{n_{\mathrm{p}}} |b_j - z_j|^2 ,$$

find an analytical *solution for* $\hat{\boldsymbol{z}}$ *in terms of* $\boldsymbol{b}$, *for the* l_1 *regularized case where* $\psi(z) = |z|$. *Sketch* $\hat{z}_j$ *vs* $b_j \in \mathbb{R}$.

- *Compare (by plotting or sketching) to the solution when an* l_2 *penalty is used where* $\psi(z) = \frac{1}{2}|z|^2$. *Repeat for at least one more of the following potential functions.*
- l_0 *potential:* $\psi(z) = \mathbb{I}_{\{z \neq 0\}}$.
- *The truncated absolute value potential:* $\psi(z) = \min(|z|, \delta)$.
- *The broken parabola potential:* $\psi(z) = \min\left(\frac{1}{2}|z|^2, \frac{1}{2}\delta^2\right)$.
- *Huber potential (1.10.9).*
- *Generalized-gaussian potential (challenging!):* $\psi(z) = |z|^p$, *for* $p \neq 1$. *(Focus on* $p \in \{1/2, 4/3, 3/2, 2, 3, 4\}$ *[205].)*
- *The hyperbola potential (challenging!):* $\psi(z) = \delta^2(\sqrt{1 + |z/\delta|^2} - 1)$.

This problem relates to wavelet-based ***denoising*** *using* ***shrinkage*** *[206] and* ***soft thresholding*** *[207], [208].*

p,res,non,line,site

Problem 1.13 *Show that when the non-interacting line-site penalty function (1.12.4) is used in the joint penalized-likelihood estimator (1.12.3), the solution to* $\hat{\boldsymbol{x}}$ *reduces to the ordinary penalized-likelihood form (1.8.1) with* $\mathsf{R}(\boldsymbol{x})$ *as in (1.10.10) and* ψ *as the* ***broken parabola*** *in (1.12.1).*

p,res,C,hv

Problem 1.14 *Let* $\boldsymbol{D}_N$ *denote the* $(N-1) \times N$ *one-dimensional finite-differencing matrix shown in (1.8.4), and* $\boldsymbol{I}_N$ *denote the* $N \times N$ *identity matrix. Show that the simple quadratic penalty (1.10.1) that uses only horizontal and vertical differences can be written in the form (1.10.7), where* $\boldsymbol{C}$ *is the following* $[N(M-1) + M(N-1)] \times MN$ *matrix:*

e,res,penal2,C,hv

$$\boldsymbol{C} = \begin{bmatrix} \boldsymbol{I}_N \otimes \boldsymbol{D}_M \\ \boldsymbol{D}_N \otimes \boldsymbol{I}_M \end{bmatrix}, \tag{1.14.2}$$

and "⊗" denotes the ***Kronecker product*** *defined in (26.1.12).*

p,res,C,kron

Problem 1.15 *For regularized restoration of a* $M \times N$ *image using a penalty function* $\mathsf{R}(\boldsymbol{x}) = \frac{1}{2}\|\boldsymbol{C}\boldsymbol{x}\|^2$, *one option is to use*

e,res,C,kron

$$\boldsymbol{C} = \begin{bmatrix} \boldsymbol{D}_{MN} \\ \boldsymbol{T} \end{bmatrix}, \tag{1.14.3}$$

where $\boldsymbol{T}$ *is an* $M(N-1) \times MN$ *Toeplitz matrix with first row* $[-1 \; \boldsymbol{0}'_{M-1} \; 1 \; \boldsymbol{0}'_{MN-M-1}]$, *where* $\boldsymbol{0}'_K$ *denotes the row vector of* K *zeros. Another option is to use* $\boldsymbol{C}$ *defined in (1.14.2). Using (1.14.3) may be slightly faster (in ANSI C). Explain the advantage of using (1.14.2).*

p,res,eig,R

Problem 1.16 *The 1D regularizer Hessian matrix in (1.8.6) has eigenvalues given in footnote 13.*
Consider the 2D regularizer $\boldsymbol{C}$ *for a* $M \times N$ *image given in (1.14.2), and define the Hessian matrix* $\boldsymbol{R} = \boldsymbol{C}'\boldsymbol{C}$. *Determine analytically the eigenvalues of* $\boldsymbol{R}$.

p,res,srp,dtft1

Problem 1.17 *Consider the discrete-space denoising problem with no boundary conditions and zero-mean white noise:*

$$g[n] = f[n] + \varepsilon[n], \quad n \in \mathbb{Z}, .$$

Analyze the spatial resolution properties of the following quadratically-regularized denoising estimator for $\beta > 0$*:*

$$\hat{f} = \mathop{\arg\min}_{f \in \ell_2} \sum_{n=-\infty}^{\infty} \frac{1}{2} |g[n] - f[n]|^2 + \beta \sum_{n=-\infty}^{\infty} \frac{1}{2} |f[n] - f[n-1]|^2 .$$

Hint. Using the DTFT, first find the frequency-domain relationship between $\mathsf{E}\left[\hat{f}\right]$ *and* f.

Optional: show that $\mathsf{E}\left[\hat{f}\right] = h * f$, *where the impulse response is*

$$h[n] = a b^{|n|} = a\, \mathrm{e}^{-|\log b||n|}, \quad b = \frac{1 + 2\beta - \sqrt{1 + 4\beta}}{2\beta} = \frac{2\beta}{1 + 2\beta + \sqrt{1 + 4\beta}},$$

where $a = \frac{1+b^2}{1-b^2} \frac{1}{1+2\beta}$. *Note that* $0 < b < 1$. *Use §27.6.1. This is one of the few cases where we can find an explicit expression for the impulse response of a regularized problem [209], [210].*
Determine the FWHM of the impulse response in terms of b.

p,res,srp,dtft2

Problem 1.18 *Generalize Problem 1.17 to the case where the regularizer is* $\sum_{n=-\infty}^{\infty} \frac{1}{2} |c[n] * f[n]|^2$, *where the filter* $c[n]$ *corresponds to* ***2nd-order finite differences****:* $c[n] = 2\,\delta[n] - \delta[n-1] - \delta[n+1]$. *This is called* ***Hodrick-Prescott*** *filtering [211] in some fields.*

p,res,srp,dtft,ma

Problem 1.19 *Generalize Problem 1.17 to the case where the regularizer is* $\sum_{n=-\infty}^{\infty} \frac{1}{2} \left| f[n] - c[n] * f[n] \right|^2$, *where the filter* $h[n]$ *is the 3-point moving average:* $c[n] = \frac{1}{3}\left(\delta[n] + \delta[n-1] + \delta[n+1]\right)$.

p,res,srp

Problem 1.20 *Analyze the spatial resolution properties of the following different denoising problems, where in each case we form an estimate* $\hat{f} = \arg\min_f \Psi(f)$.

- *Continuous-continuous formulation (use Fourier transform):*

$$\Psi(f) = \frac{1}{2} \|g - f\|^2 + \beta \int \frac{1}{2} \left|\dot{f}\right|^2$$

- *Continuous-discrete formulation (use Fourier series?):*

$$\Psi(f) = \sum_{n=0}^{N-1} \frac{1}{2} \left|g_n - f(n/N)\right|^2 + \beta \int_0^1 \frac{1}{2} \left|\dot{f}\right|^2 .$$

In each case one can find an expression similar to (1.9.1) or (1.9.2).
Hint. Yet another formulation is the finite-length discrete case with periodic end conditions:

$$\Psi(f) = \sum_{n=0}^{N-1} \frac{1}{2} \left|g[n] - f[n]\right|^2 + \beta \sum_{n=0}^{N-1} \frac{1}{2} \left|f[n] - f[n \bmod N]\right|^2 .$$

Letting F_k *denote the* N*-point DFT of* $f[n]$*, by* ***Parseval's relation*** *for the DFT, we can express the cost function as*

$$\sum_{n=0}^{N-1} \frac{1}{2} \left|G_k - F_k\right|^2 + \beta \sum_{n=0}^{N-1} \frac{1}{2} \left|F_k - \mathrm{e}^{-\imath 2\pi k/N} F_k\right|^2 = \sum_{n=0}^{N-1} \frac{1}{2} \left|G_k - F_k\right|^2 + \beta \sum_{n=0}^{N-1} \mathsf{R}_k \frac{1}{2} \left|F_k\right|^2 ,$$

where $\mathsf{R}_k = \left|1 - \mathrm{e}^{-\imath 2\pi k/N}\right|^2 = 2 - 2\cos(2\pi k/N)$. *Differentiating w.r.t.* F_k *and equating to zero yields* $\hat{F}_k = L_k G_k$*, where* L_k *was defined in (1.8.16), so the resolution properties here are* $\mathsf{E}\left[\hat{F}_k\right] = \frac{1}{1+\beta \mathsf{R}_k} F_k$.

p,res,srp,cs1d

Problem 1.21 *Consider the 1D continuous-space denoising problem* $g(t) = f(t) + \varepsilon(t)$ *with estimator*

$$\hat{f} = \arg\min_f \Psi(f), \qquad \Psi(f) = \int \frac{1}{2} \left|g(t) - f(t)\right|^2 \mathrm{d}t + \beta \int \frac{1}{2} \left|f^{(p)}(t)\right|^2 \mathrm{d}t,$$

where $f^{(p)}$ *denotes the pth derivative of* f*. Analyze the spatial resolution properties of the estimator* $\hat{f}$*. (i) Find a general expression for the frequency response, akin to (1.9.2). (ii) Using [213] and/or §27.4.1, verify the following specific expressions for the impulse response for* $p = 1$ *(1st-order regularization) and* $p = 2$ *(2nd-order regularization):*

$$h_1(t) = \frac{1}{2\sqrt{\beta}} \mathrm{e}^{-|t|/\sqrt{\beta}}, \qquad h_2(t) = \frac{1}{2\beta^{1/4}} \mathrm{e}^{-|t|/(\sqrt{2}\beta^{1/4})} \sin\!\left(\frac{|t|}{\sqrt{2}\beta^{1/4}} + \frac{\pi}{4}\right).$$

(iii) Show also that $\text{FWHM}_1 = (2\log_2)\beta^{1/2}$ *and* $\text{FWHM}_2 \propto \beta^{1/4}$*. (iv) Use the plot in Fig. 1.14.1 of these two PSFs (at matched FWHM) to compare them qualitatively. (v) For even* $p \geq 2$ *and* $\beta = 1$*, use §27.4.1 to find a general expression for the impulse response of the following form:*

$$h_p(t) = \frac{1}{p} \sum_{l=0}^{p/2-1} \left[\frac{b_l^2}{a_l} \cos(b_l t) + \frac{a_l^2}{b_l} \sin(b_l |t|)\right] \mathrm{e}^{-a_l |t|} . \tag{1.14.4}$$

e,res,srp,cs1d,hp

p,res,up2

Problem 1.22 *Consider the 1D "regularized interpolation" or "super-resolution" problem, where we measure the even (or odd) samples of a 1D signal* $x[n]$ *with noise:*

$$y[n] = x[2n] + \varepsilon[n],$$

and we wish to recover $x[\cdot]$ *by a quadratically penalized LS method:*

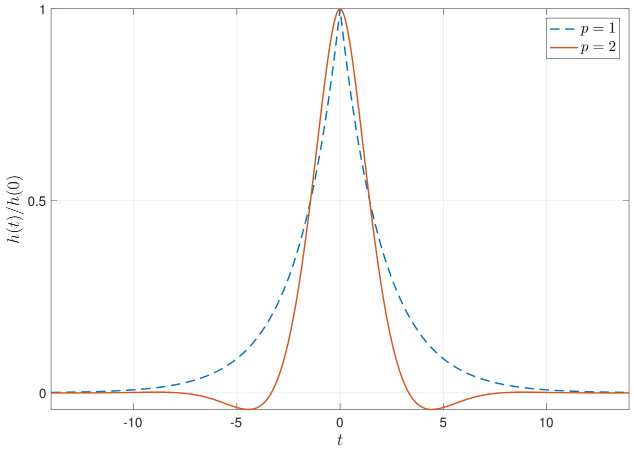
$$\hat{x} = \arg\min_x \sum_{n=-\infty}^{\infty} \frac{1}{2} \left|y[n] - x[2n]\right|^2 + \beta \sum_{n=-\infty}^{\infty} \frac{1}{2} \left|(c * x)[n]\right|^2 ,$$

where $c[n]$ *is a (typically high-pass) filter. We want to determine the frequency-domain relationship between* $\hat{x}[n]$ *and* $y[n]$*. Towards this end, consider the model* $y_k[n] = s_k[n] + \varepsilon_k[n]$ *where for* $k = 0, 1$*:*

$$s_k[n] = \frac{1 + (-1)^k (-1)^n}{2} x[n] \overset{\text{DTFT}}{\longleftrightarrow} S_k(\Omega) = \frac{X(\Omega) + (-1)^k X(\Omega \pm \pi)}{2}.$$

So $y_0[n]$ *is the even samples and* $y_1[n]$ *is the odd samples of* $x[n]$ *plus noise. Let*

$$\hat{x}_k = \arg\min_x \sum_n \left|y_k[n] - \frac{1 + (-1)^k (-1)^n}{2} x[n]\right|^2 + \beta \sum_n \left|(c * x)[n]\right|^2 .$$



fig_prob_res_srp_cs1d

Figure 1.14.1: PSFs of 1st-order and 2nd-order regularization for 1D continuous-space denoising.

- *Determine the frequency domain relationship between $\hat{x}_k[n]$ and $y_k[n]$.*
- *Does regularization overcome the possible aliasing associated with down sampling a signal?*

p,res,up2,psf

Problem 1.23 *Continuing Problem 1.22, analyze the resolution properties. Let $\hat{x} \triangleq \frac{1}{2}(\hat{x}_0 + \hat{x}_1)$, and show that* $\mathsf{E}\left[\hat{X}(\Omega)\right] = L(\Omega)X(\Omega)$ *where* $L(\Omega) = \dfrac{R(\Omega \pm \pi)}{R(\Omega) + R(\Omega \pm \pi) + 2\beta R(\Omega)R(\Omega \pm \pi)}$.

p,res,up3

Problem 1.24 *Generalizing Problem 1.22, consider a 1D "super-resolution" problem where we measure every 3rd sample of a blurred 1D signal $x[n], n = 0, \ldots, N-1$ with noise:*

$$y[m] = (b * x)[3m] + \varepsilon[m], \quad m = 0, \ldots, M-1,$$

where in this problem we consider periodic convolution throughout and $N = 3M$. Here we recover $x[\cdot]$ by a quadratically penalized LS method expressed in a matrix formulation:

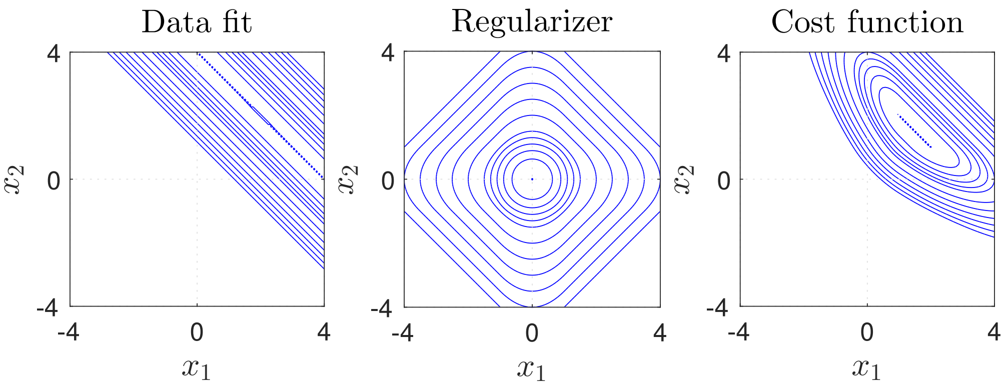
$$\hat{\boldsymbol{x}} = \arg\min_{\boldsymbol{x}} \frac{1}{2} \|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}\|_2^2 + \frac{1}{2} \|\boldsymbol{C}\boldsymbol{x}\|_2^2,$$

where $\boldsymbol{C}'\boldsymbol{C}$ is a circulant matrix, typically representing a high-pass filter, and $\boldsymbol{A} = \boldsymbol{S}\boldsymbol{B}$ is a $M \times N$ matrix where $\boldsymbol{B}$ is a $N \times N$ circulant matrix corresponding to the blur $b[n]$ and $\boldsymbol{S} = \boldsymbol{I}_N \otimes [1\ 0\ 0]$ is a $M \times N$ "down-sampling" matrix. Find a matrix expression for the solution and show how one can use FFT's to implement that solution efficiently. Note that $\boldsymbol{A}$ is not *circulant (it is rectangular) nor is $\boldsymbol{A}'\boldsymbol{A}$ is circulant.*

p,res,convex1

Problem 1.25 *This problem considers whether the penalized least-squares cost function $\Psi(\boldsymbol{x})$ in (1.10.11) has a* ***unique minimizer*** *in the usual cases where $\boldsymbol{A}$ and $\boldsymbol{C}$ have* ***disjoint null spaces****.*

- *Prove that if the potential function ψ used in (1.10.11) is twice differentiable with a positive second derivative, then Ψ is strictly convex (and thus has a unique minimizer).*
- *What if ψ is strictly convex, but does not necessarily have a positive second derivative? An example would be $\psi(z) = |z|^4$.*
- *What if ψ is merely convex, like the Huber function? Hint: see Fig. 1.14.2.*



fig_prob_res_convex1

Figure 1.14.2: Contours of data fit term, regularizer, and cost function $\Psi(\boldsymbol{x})$ for Problem 1.25.

p,res,fixed

Problem 1.26 *Consider the regularized least-squares problem (1.10.11) with regularizer (1.10.10) and the usual 1st-order finite differencing matrix* $\boldsymbol{C}$.

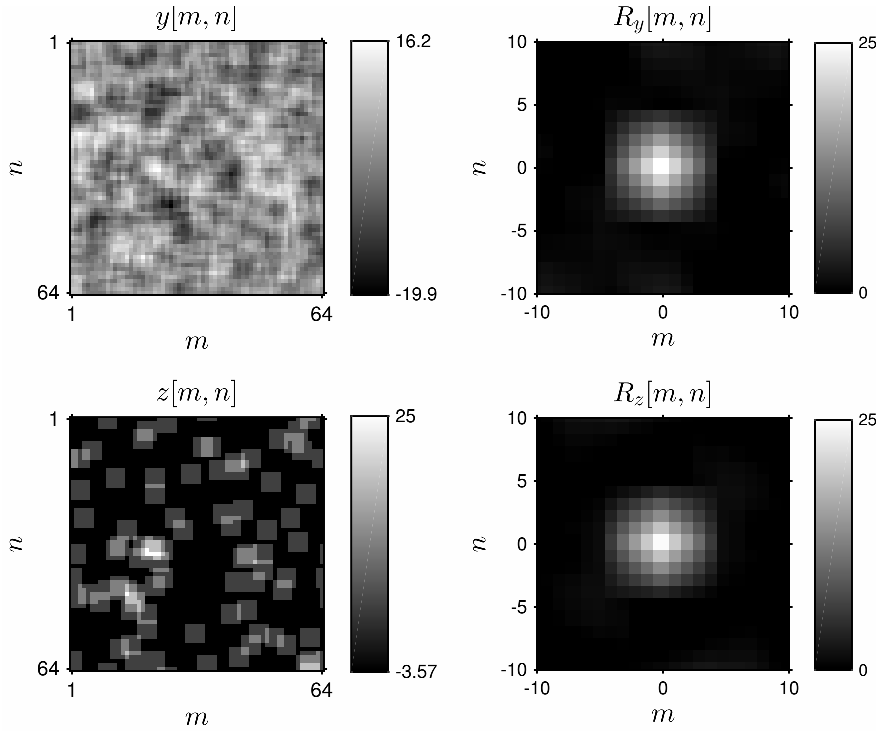
- *Ken uses the generalized-gaussian potential function* $\psi(z) = |z|^q$ *with* $q = 1.5$, *and states that the solution* $\hat{\boldsymbol{x}}$ *satisfies the recursive expression (1.10.16). Discuss.*
- *Maria uses the Geman & McClure potential function* $\psi(z) = |z|^2 / (1 + |z|^2)$ *and also states that the solution* $\hat{\boldsymbol{x}}$ *satisfies the recursive expression (1.10.16). Discuss.*

p,res,grf2

Problem 1.27 *Use circulant end conditions to synthesize a gaussian random field image like that in Fig. 1.7.1, for the 2D finite differencing matrix* $\boldsymbol{C}$ *defined in (1.10.8).*
Hint. The goal is to draw $\boldsymbol{x} \sim \mathsf{N}(\boldsymbol{0}, \boldsymbol{K_x})$, *so let* $\boldsymbol{x} = \boldsymbol{K}_{\boldsymbol{x}}^{1/2} \boldsymbol{w}$ *where* $\boldsymbol{w} \sim \mathsf{N}(\boldsymbol{0}, \boldsymbol{I})$. *In this case,* $\boldsymbol{K_x} = [\boldsymbol{C}'\boldsymbol{C}]^{-1}$. *Because* $\boldsymbol{K_x}$ *is not invertible, use its pseudo inverse. Do* not *use the* `pinv` *command; use FFTs.*

p,res,auto2

Problem 1.28 *Any (stationary)* ***gaussian random field*** *(**GRF**) is defined solely by its mean and autocorrelation function. Fig. 1.14.3 illustrates that the autocorrelation of a random field provides limited insight into its nature. The random process* $y[m,n]$ *was generated by filtering white noise with the filter* $h_1[m,n] = \mathbb{I}_{\{|n|\leq 2\}}\mathbb{I}_{\{|m|\leq 2\}}$. *Random process* $z[m,n]$ *was generated by first defining a Bernoulli random field that takes the value* $1-p$ *with probability* p *and the value* $-p$ *with probability* $1-p$ *for some* $p \in (0,1)$, *and then convolving that field with the filter* $h_2[m,n] = h_1[m,n] / \sqrt{p(1-p)}$. *Show that* $y[m,n]$ *and* $z[m,n]$ *have identical mean and autocorrelation functions.*



fig_prob_res_auto2

Figure 1.14.3: Two different wide-sense stationary random fields with identical mean and autocorrelation functions.

p,res,psf,noise

Problem 1.29 *Consider a 1D deconvolution problem with blur* $b[n] = \frac{1}{4}\delta[n+1] + \frac{1}{2}\delta[n] + \frac{1}{4}\delta[n-1]$. *Assuming periodic end conditions for* $N = 64$, *use the circulant analyses of §1.8.3 and §1.9 to make a single plot showing the trade-off between spatial resolution (such as FWHM of the PSF) and noise (*$\mathsf{Var}\{\hat{x}_j\} / \sigma^2$*) over a range of values of* β. *(Horizontal axis should be FWHM; vertical axis should be noise standard deviation:* $\sqrt{\mathsf{Var}\{\hat{x}_j\}}$. *Plot curves for regularization based on both first- and second-order finite differences and compare.*

p,res,icm

Problem 1.30 *The* ***iterated conditional modes*** *(**ICM**) algorithm [214] for MAP estimation from the model* $\boldsymbol{y} = \boldsymbol{x} + \boldsymbol{\varepsilon}$ *uses the following recursion:*

$$x_j^{(n+1)} = \arg\max_{x_j} \mathsf{p}\left(x_j \,|\, y_j, \boldsymbol{x}_{\mathcal{N}_j}^{(n)}\right) = \arg\max_{x_j} \left[\log(\mathsf{p}(y_j \,|\, x_j)) + \log\left(\mathsf{p}\left(x_j \,|\, \boldsymbol{x}_{\mathcal{N}_j}^{(n)}\right)\right)\right],$$

where $\mathcal{N}_j$ *denotes the neighborhood of the* j*th pixel.*
Derive the ICM algorithm for the case where $\boldsymbol{\varepsilon}$ *is zero-mean white gaussian noise with variance* σ^2 *and where the MRF prior has (improper) gaussian distribution:* $\mathsf{p}(\boldsymbol{x}) = \frac{1}{Z} \mathrm{e}^{-\frac{\beta}{2}\mathsf{R}(\boldsymbol{x})}$ *with* $\mathsf{R}(\boldsymbol{x}) = \sum_{n=0}^{N-1} \frac{1}{2} |x[n] - x[(n-1) \bmod N]|^2 = \frac{1}{2}\|\boldsymbol{C}\boldsymbol{x}\|^2$ *where* $\boldsymbol{C}$ *is the circulant finite-differencing matrix defined in (1.8.12). Show that this ICM algorithm is convergent by relating it to a* ***gradient descent*** *(**GD**) algorithm with a suitable step size.*

p,res,fir,wiener

Problem 1.31 *From §1.8.3, image denoising with quadratic regularizer (1.8.11) is equivalent to applying a low-pass Wiener-like filter (1.8.16) with frequency response*

$$L_{\text{IIR}}(\Omega) = \frac{1}{1+\beta R(\Omega)}, \quad R(\Omega) = 2 - 2\cos(\Omega)\,.$$

This is an ***IIR filter*** *and in some situations it may be preferable to use a* ***FIR filter****. Rearranging, we can solve for* $\beta R(\Omega)$ *in terms of* $L(\Omega)$ *to design a IIR regularizer for a given desired response* $L(\Omega)$*:*

$$\beta R(\Omega) = \frac{1}{L(\Omega)} - 1.$$

Because quadratic regularization blurs edges, often we would use a small value of β*, so the FWHM of the low-pass filter is at most 2 or 3 pixels, in which case a FIR filter need only have a few taps, for which*

$$L_{\text{FIR}}(\Omega) = b_0 + 2b_1\cos(\Omega) + 2b_2\cos(2\Omega)\,.$$

Determine the values of the FIR filter taps b_0, b_1, b_2 *so that*
$L_{\text{FIR}}(0) = L_{\text{IIR}}(0)$, $L_{\text{FIR}}(\pi) = L_{\text{IIR}}(\pi)$, *and* $\frac{\mathrm{d}}{\mathrm{d}\Omega} L_{\text{FIR}}(\Omega) \leq 0,\ 0 \leq \Omega \leq \pi$.

Problem 1.32 *Prove or disprove whether the following two sparsity formulations are equivalent if* $\boldsymbol{B}$ *is a* ***tight frame****:*

$$\textit{(restricted) synthesis: } \hat{\boldsymbol{x}} = \boldsymbol{B}\hat{\boldsymbol{\theta}}, \quad \hat{\boldsymbol{\theta}} = \arg\min_{\boldsymbol{\theta}} \|\boldsymbol{\theta}\| \text{ sub. to } \|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{B}\boldsymbol{\theta}\| < \varepsilon \textit{ and } \boldsymbol{B}'\boldsymbol{B}\boldsymbol{\theta} = \boldsymbol{\theta}$$

$$\textit{analysis: } \hat{\boldsymbol{x}} = \arg\min_{\boldsymbol{x}} \|\boldsymbol{B}'\boldsymbol{x}\| \text{ sub. to } \|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}\| < \varepsilon.$$

(Solve?)

1.15 Bibliography

iinuma:67:iri [1] T. A. Iinuma and T. Nagai, "Image restoration in radioisotopic imaging systems," *Phys. Med. Biol.*, vol. 12, no. 4, 501–510, Oct. 1967. DOI: 10.1088/0031-9155/12/4/005 (cit. on p. 1.2).

andrews:77 [2] H. C. Andrews and B. R. Hunt, *Digital image restoration*. NJ: Prentice-Hall, 1977 (cit. on p. 1.2).

bates:86 [3] R. H. T. Bates and M. J. McDonnell, *Image restoration and reconstruction*. New York: Oxford, 1986 (cit. on p. 1.2).

lagendijk:91 [4] R. L. Lagendijk and J. Biemond, *Iterative identification and restoration of images*. Boston: Kluwer, 1991 (cit. on p. 1.2).

sezan:90:sor [5] M. I. Sezan and A. M. Tekalp, "Survey of recent developments in digital image restoration," *Optical Engineering*, vol. 29, no. 5, 393–404, May 1990. DOI: 10.1117/12.55610 (cit. on p. 1.2).

johnson:06:ira [6] V. E. Johnson, "Image restoration and reconstruction," in *Encyclopedia of Statistical Sciences*, Wiley, 2006. DOI: 10.1002/9781118445112.stat00051 (cit. on p. 1.2).

snyder:93:irf [7] D. L. Snyder, A. M. Hammoud, and R. L. White, "Image recovery from data acquired with a charge-coupled-device camera," *J. Opt. Soc. Am. A*, vol. 10, no. 5, 1014–23, May 1993. DOI: 10.1364/JOSAA.10.001014 (cit. on pp. 1.2, 1.17).

snyder:95:cfr [8] D. L. Snyder, C. W. Helstrom, A. D. Lanterman, M. Faisal, and R. L. White, "Compensation for readout noise in CCD images," *J. Opt. Soc. Am. A*, vol. 12, no. 2, 272–83, Feb. 1995. DOI: 10.1364/JOSAA.12.000272 (cit. on pp. 1.2, 1.17).

llacer:90:iml [9] J. Llacer and J. Núñez, "Iterative maximum likelihood and Bayesian algorithms for image reconstruction in astronomy," in *Restoration of Hubble Space Telescope Images*, R. L. White and R. J. Allen, Eds., Baltimore: Space telescope science institute, 1990, pp. 62–9 (cit. on p. 1.2).

sotthivirat:02:iru [10] S. Sotthivirat and J. A. Fessler, "Image recovery using partitioned-separable paraboloidal surrogate coordinate ascent algorithms," *IEEE Trans. Im. Proc.*, vol. 11, no. 3, 306–17, Mar. 2002. DOI: 10.1109/83.988963 (cit. on p. 1.2).

galatsanos:89:dro [11] N. P. Galatsanos and R. T. Chin, "Digital restoration of multichannel images," *IEEE Trans. Sig. Proc.*, vol. 37, no. 3, 415–21, Mar. 1989. DOI: 10.1109/29.21708 (cit. on p. 1.2).

berenstein:90:edf [12] C. A. Berenstein and E. V. Patrick, "Exact deconvolution for multiple convolution operators-an overview, plus performance characterizations for imaging sensors," *Proc. IEEE*, vol. 78, no. 4, 723–34, Apr. 1990. DOI: 10.1109/5.54810 (cit. on pp. 1.2, 1.4).

galatsanos:91:lsr [13] N. P. Galatsanos, A. K. Katsaggelos, R. T. Chin, and A. D. Hillery, "Least squares restoration of multichannel images," *IEEE Trans. Sig. Proc.*, vol. 39, no. 10, 2222–36, Oct. 1991. DOI: 10.1109/78.91180 (cit. on p. 1.2).

schultz:95:sma [14] R. R. Schultz and R. L. Stevenson, "Stochastic modeling and estimation of multispectral image data," *IEEE Trans. Im. Proc.*, vol. 4, no. 8, 1109–19, Aug. 1995. DOI: 10.1109/83.403416 (cit. on p. 1.2).

cannon:76:bdo [15] M. Cannon, "Blind deconvolution of spatially invariant image blurs with phase," *IEEE Trans. Acoust. Sp. Sig. Proc.*, vol. 24, no. 1, 58–63, Feb. 1976. DOI: 10.1109/TASSP.1976.1162770 (cit. on p. 1.4).

lane:87:amd [16] R. G. Lane and R. H. T. Bates, "Automatic multidimensional deconvolution," *J. Opt. Soc. Am. A*, vol. 4, no. 1, 180–8, Jan. 1987. DOI: 10.1364/JOSAA.4.000180 (cit. on p. 1.4).

ayers:88:ibd [17] G. R. Ayers and J. C. Dainty, "Iterative blind deconvolution method and its applications," *Optics Letters*, vol. 13, no. 7, 547–9, Jul. 1988. DOI: 10.1364/OL.13.000547 (cit. on p. 1.4).

lay:90:iia [18] K.-T. Lay and A. K. Katsaggelos, "Image identification and restoration based on the expectation-maximization algorithm," *Optical Engineering*, vol. 29, no. 5, 436–45, May 1990. DOI: 10.1117/1.2168789 (cit. on p. 1.4).

tong:91:iai [19] L. Tong, R. Liu, V. C. Soon, and Y. Huang, "Indeterminacy and identifiability of blind identification," *IEEE Trans. Circ. Sys.*, vol. 38, no. 5, 499–509, May 1991. DOI: 10.1109/31.76486 (cit. on p. 1.4).

holmes:92:bdo [20] T. J. Holmes, "Blind deconvolution of quantum-limited incoherent imagery: Maximum-likelihood approach," *J. Opt. Soc. Am. A*, vol. 9, no. 7, 1052–61, Jul. 1992. DOI: 10.1364/JOSAA.9.001052 (cit. on p. 1.4).

lane:92:bdo [21] R. G. Lane, "Blind deconvolution of speckle images," *J. Opt. Soc. Am. A*, vol. 9, no. 9, 1508–, Sep. 1992. DOI: 10.1364/JOSAA.9.001508 (cit. on p. 1.4).

reeves:92:bib [22] S. J. Reeves and R. M. Mersereau, "Blur identification by the method of generalized cross-validation," *IEEE Trans. Im. Proc.*, vol. 1, no. 3, 301–11, Jul. 1992. DOI: 10.1109/83.148604 (cit. on p. 1.4).

fish:95:bdb [23] D. A. Fish, A. M. Brinicombe, E. R. Pike, and J. G. Walker, "Blind deconvolution by means of the Richardson-Lucy algorithm," *J. Opt. Soc. Am. A*, vol. 12, no. 1, 58–65, Jan. 1995. DOI: 10.1364/JOSAA.12.000058 (cit. on p. 1.4).

thiebaut:95:sap [24] E. Thiébaut and J. M. Conan, "Strict a priori constraints for maximum-likelihood blind deconvolution," *J. Opt. Soc. Am. A*, vol. 12, no. 3, 485–92, Mar. 1995. DOI: 10.1364/JOSAA.12.000485 (cit. on p. 1.4).

kundur:96:bid [25] D. Kundur and D. Hatzinakos, "Blind image deconvolution," *IEEE Sig. Proc. Mag.*, vol. 13, no. 3, 43–64, May 1996. DOI: 10.1109/79.489268 (cit. on p. 1.4).

you:96:ara [26] Y.-L. You and M. Kaveh, "A regularization approach to joint blur identification and image restoration," *IEEE Trans. Im. Proc.*, vol. 5, no. 3, 416–28, Mar. 1996. DOI: 10.1109/83.491316 (cit. on p. 1.4).

conan:98:mdo [27] J.-M. Conan, L. M. Mugnier, T. Fusco, V. Michau, and G. Rousset, "Myopic deconvolution of adaptive optics images by use of object and point-spread function power spectra," *Appl. Optics*, vol. 37, no. 21, 4614–22, 1998. DOI: 10.1364/AO.37.004614 (cit. on p. 1.4).

harikumar:99:eid [28] G. Harikumar and Y. Bresler, "Exact image deconvolution from multiple FIR blurs," *IEEE Trans. Im. Proc.*, vol. 8, no. 6, 846–62, Jun. 1999. DOI: 10.1109/83.766861 (cit. on p. 1.4).

markham:99:pbd [29] J. Markham and José.-Angel. Conchello, "Parametric blind deconvolution: A robust method for the simultaneous estimation of image and blur," *J. Opt. Soc. Am. A*, vol. 16, no. 10, 2377–91, Oct. 1999. DOI: 10.1364/JOSAA.16.002377 (cit. on p. 1.4).

chan:00:cot [30] T. F. Chan and C. K. Wong, "Convergence of the alternating minimization algorithm for blind deconvolution," *Linear Algebra and its Applications*, vol. 316, no. 1, 259–85, Sep. 2000. DOI: 10.1016/S0024-3795(00)00141-5 (cit. on p. 1.4).

lam:00:isa [31] E. Y. Lam and J. W. Goodman, "Iterative statistical approach to blind image deconvolution," *J. Opt. Soc. Am. A*, vol. 17, no. 7, 1177–84, Jul. 2000. DOI: 10.1364/JOSAA.17.001177 (cit. on p. 1.4).

mesarovic:00:ilm [32] V. Z. Mesarovic, N. P. Galatsanos, and M. N. Wernick, "Iterative linear minimum mean-square-error image restoration from partially known blur," *J. Opt. Soc. Am. A*, vol. 17, no. 4, 711–23, Apr. 2000. DOI: 10.1364/JOSAA.17.000711 (cit. on p. 1.4).

efromovich:01:oip [33] S. Efromovich and V. Koltchinskii, "On inverse problems with unknown operators," *IEEE Trans. Info. Theory*, vol. 47, no. 7, 2876–94, Nov. 2001. DOI: 10.1109/18.959267 (cit. on p. 1.4).

yap:03:ars [34] K.-H. Yap, L. Guan, and W. Liu, "A recursive soft-decision approach to blind image deconvolution," *IEEE Trans. Sig. Proc.*, vol. 51, no. 2, 515–26, Feb. 2003. DOI: 10.1109/TSP.2002.806985 (cit. on p. 1.4).

justen:06:ani [35] L. Justen and R. Ramlau, "A non-iterative regularization approach to blind deconvolution," *Inverse Prob.*, vol. 22, no. 3, 771–800, Jun. 2006. DOI: 10.1088/0266-5611/22/3/003 (cit. on p. 1.4).

labat:06:sbd [36] C. Labat and Jérôme. Idier, "Sparse blind deconvolution accounting for time-shift ambiguity," in *Proc. IEEE Conf. Acoust. Speech Sig. Proc.*, vol. 3, 2006, 616–9. DOI: 10.1109/ICASSP.2006.1660729 (cit. on p. 1.4).

herrity:08:bro [37] K. Herrity, R. Raich, and A. O. Hero, "Blind reconstruction of sparse images with unknown point spread function," in *Proc. SPIE 6814 Computational Imaging VI*, 2008, 68140K. DOI: 10.1117/12.779253 (cit. on p. 1.4).

schulz:93:mbd [38] T. J. Schulz, "Multiframe blind deconvolution of astronomical images," *J. Opt. Soc. Am. A*, vol. 10, no. 5, 1064–73, May 1993. DOI: 10.1364/JOSAA.10.001064 (cit. on p. 1.4).

casey:94:soc [39] S. D. Casey and D. F. Walnut, "Systems of convolution equations, deconvolution, Shannon sampling, and the wavelet and Gabor transforms," *SIAM Review*, vol. 36, no. 4, 537–77, Dec. 1994. DOI: 10.1137/1036140 (cit. on p. 1.4).

pai:97:emb [40] H.-T. Pai and A. C. Bovik, "Exact multichannel blind image restoration," *IEEE Signal Proc. Letters*, vol. 4, no. 8, 217–20, Aug. 1997. DOI: 10.1109/97.611280 (cit. on p. 1.4).

pai:98:mbi [41] H. T. Pai, A. C. Bovik, and B. L. Evans, "Multichannel blind image restoration," *Elektrik*, vol. 5, no. 1, 79–97, 1998 (cit. on p. 1.4).

tong:98:mbi [42] L. Tong and S. Perreau, "Multichannel blind identification: From subspace to maximum likelihood methods," *Proc. IEEE*, vol. 86, no. 10, 1951–68, Oct. 1998. DOI: 10.1109/5.720247 (cit. on p. 1.4).

harikumar:99:pbr [43] G. Harikumar and Y. Bresler, "Perfect blind restoration of images blurred by multiple filters: Theory and efficient algorithms," *IEEE Trans. Im. Proc.*, vol. 8, no. 2, 202–19, Feb. 1999. DOI: 10.1109/83.743855 (cit. on p. 1.4).

wernecke:77:mei [44] S. J. Wernecke and L. R. DAddario, "Maximum entropy image reconstruction," *IEEE Trans. Comp.*, vol. 26, no. 4, 351–64, Apr. 1977. DOI: 10.1109/TC.1977.1674845 (cit. on pp. 1.4, 1.36).

natterer:01 [45] F. Natterer and F. Wübbeling, *Mathematical methods in image reconstruction*. Philadelphia: Soc. Indust. Appl. Math., 2001. DOI: 10.1137/1.9780898719284 (cit. on p. 1.4).

aubert:06 [46] G. Aubert and P. Kornprobst, *Mathematical problems in image processing: PARTIAL differential equations and the calculus of variations*, 2nd ed. Springer, 2006. [Online]. Available: http://www-sop.inria.fr/books/imath/ (cit. on p. 1.4).

luenberger:69 [47] D. G. Luenberger, *Optimization by vector space methods*. New York: Wiley, 1969. [Online]. Available: http://books.google.com/books?id=lZU0CAH4RccC (cit. on pp. 1.5, 1.36).

bertero:85:lip [48] M. Bertero, C. De Mol, and E. R. Pike, “Linear inverse problems with discrete data, I: General formulation and singular system analysis,” *Inverse Prob.*, vol. 1, no. 4, 301–30, Nov. 1985. DOI: 10.1088/0266-5611/1/4/004 (cit. on p. 1.5).

kybic:01:vat [49] J. Kybic, T. Blu, and M. A. Unser, “Variational approach to tomographic reconstruction,” in *Proc. SPIE 4322 Medical Imaging 2001: Image Proc.*, vol. 1, 2001, 30–9. DOI: 10.1117/12.431108 (cit. on p. 1.5).

kybic:02:gsa-1 [50] J. Kybic, T. Blu, and M. Unser, “Generalized sampling: A variational approach—I. Theory,” *IEEE Trans. Sig. Proc.*, vol. 50, no. 8, 1965–76, Aug. 2002. DOI: 10.1109/TSP.2002.800391 (cit. on p. 1.5).

kybic:02:gsa-2 [51] ——, “Generalized sampling: A variational approach—II. Applications,” *IEEE Trans. Sig. Proc.*, vol. 50, no. 8, 1977–85, Aug. 2002. DOI: 10.1109/TSP.2002.800386 (cit. on p. 1.5).

bracewell:95 [52] R. N. Bracewell, *Two-dimensional imaging*. New York: Prentice-Hall, 1995 (cit. on p. 1.5).

gray:06:tac [53] R. M. Gray, *Toeplitz and circulant matrices: A review*, 2006. [Online]. Available: http://www-ee.stanford.edu/~gray/toeplitz.html (cit. on p. 1.7).

chan:07 [54] R. H.-F. Chan and X.-Q. Jin, *An introduction to iterative Toeplitz solvers*. Philadelphia: Soc. Indust. Appl. Math., 2007. [Online]. Available: http://www.ec-securehost.com/SIAM/FA05.html (cit. on pp. 1.7, 1.11).

press:92 [55] W. H. Press, B. P. Flannery, S. A. Teukolsky, and W. T. Vetterling, *Numerical recipes in C*, 2nd ed. New York: Cambridge Univ. Press, 1992 (cit. on p. 1.7).

defrise:00:ast [56] M. Defrise, Frédéric. Noo, and H. Kudo, “A solution to the long-object problem in helical cone-beam tomography,” *Phys. Med. Biol.*, vol. 45, no. 3, 623–43, Mar. 2000. DOI: 10.1088/0031-9155/45/3/305 (cit. on p. 1.8).

reeves:02:fro [57] S. J. Reeves, “Fast restoration of PMMW imagery without boundary artifacts,” in *Proc. SPIE 4719 Infrared and passive millimeter-wave imaging systems: Design, analysis, modeling, and testing*, 2002, 289–95. DOI: 10.1117/12.477469 (cit. on p. 1.8).

reeves:05:fir [58] ——, “Fast image restoration without boundary artifacts,” *IEEE Trans. Im. Proc.*, vol. 14, no. 10, 1448–53, Oct. 2005. DOI: 10.1109/TIP.2005.854474 (cit. on p. 1.8).

matakos:13:aep [59] A. Matakos, S. Ramani, and J. A. Fessler, “Accelerated edge-preserving image restoration without boundary artifacts,” *IEEE Trans. Im. Proc.*, vol. 22, no. 5, 2019–29, May 2013. DOI: 10.1109/TIP.2013.2244218 (cit. on p. 1.8).

almeida:13:diw [60] M. S. C. Almeida and M. A. T. Figueiredo, “Deconvolving images with unknown boundaries using the alternating direction method of multipliers,” *IEEE Trans. Im. Proc.*, vol. 22, no. 8, 3074–86, Aug. 2013. DOI: 10.1109/TIP.2013.2258354 (cit. on p. 1.8).

simoes:16:agf [61] M. Simões, L. B. Almeida, José. Bioucas-Dias, and J. Chanussot, *A general framework for fast image deconvolution with incomplete observations. applications to unknown boundaries, inpainting, superresolution, and demosaicing*, arxiv 1602.01410, 2016. [Online]. Available: http://arxiv.org/abs/1602.01410 (cit. on p. 1.8).

unser:99:sap [62] M. Unser, “Splines: A perfect fit for signal and image processing,” *IEEE Sig. Proc. Mag.*, vol. 16, no. 6, 22–38, Nov. 1999. DOI: 10.1109/79.799930 (cit. on p. 1.8).

ng:99:afa [63] M. Ng, R. Chan, and W. Tang, “A fast algorithm for deblurring models with Neumann boundary conditions,” *Siam-js*, vol. 21, no. 3, 851–66, 1999. DOI: 10.1137/S1064827598341384 (cit. on p. 1.8).

donatelli:05:arb [64] M. Donatelli and S. Serra-Capizzano, “Anti-reflective boundary conditions and re-blurring,” *Inverse Prob.*, vol. 21, no. 1, 169–82, Feb. 2005. DOI: 10.1088/0266-5611/21/1/011 (cit. on p. 1.9).

donatelli:06:iid [65] M. Donatelli, C. Estatico, A. Martinelli, and S. Serra-Capizzano, “Improved image deblurring with anti-reflective boundary conditions and re-blurring,” *Inverse Prob.*, vol. 22, no. 6, 2035–54, Dec. 2006. DOI: 10.1088/0266-5611/22/6/008 (cit. on p. 1.9).

donatelli:13:fnp [66] M. Donatelli and M. Hanke, “Fast nonstationary preconditioned iterative methods for ill-posed problems, with application to image deblurring,” *Inverse Prob.*, vol. 29, no. 9, p. 095 008, Sep. 2013. DOI: 10.1088/0266-5611/29/9/095008 (cit. on p. 1.9).

fan:11:sbc [67] Y. W. Fan and J. G. Nagy, “Synthetic boundary conditions for image deblurring,” *Linear Algebra and its Applications*, vol. 434, no. 11, 2244–68, Jun. 2011. DOI: 10.1016/j.laa.2009.12.021 (cit. on p. 1.9).

calvetti:05:seo [68] D. Calvetti and E. Somersalo, “Statistical elimination of boundary artefacts in image deblurring,” *Inverse Prob.*, vol. 21, no. 5, 1697–714, Oct. 2005. DOI: 10.1088/0266-5611/21/5/012 (cit. on p. 1.11).

cahill:70:tpa [69] P. Cahill and L. M. Blau, “The preliminary application of a matrix inversion method for radionuclide imaging,” *J. Nuc. Med.*, vol. 11, no. 10, 613–5, 1970. [Online]. Available: http://jnm.snmjournals.org/cgi/content/abstract/11/10/613 (cit. on p. 1.14).

yavuz:98:sir [70] M. Yavuz and J. A. Fessler, “Statistical image reconstruction methods for randoms-precorrected PET scans,” *Med. Im. Anal.*, vol. 2, no. 4, 369–78, Dec. 1998. DOI: 10.1016/S1361-8415(98)80017-0 (cit. on p. 1.17).

yavuz:99:ple [71] ——, “Penalized-likelihood estimators and noise analysis for randoms-precorrected PET transmission scans,” *IEEE Trans. Med. Imag.*, vol. 18, no. 8, 665–74, Aug. 1999. DOI: 10.1109/42.796280 (cit. on p. 1.17).

whiting::mxr [72] B. R. Whiting, L. J. Montagnino, and D. G. Politte, *Modeling X-ray computed tomography sinograms*, submitted to mp, 2001 (cit. on p. 1.17).

lucy:74:ait [73] L. Lucy, “An iterative technique for the rectification of observed distributions,” *The Astronomical Journal*, vol. 79, no. 6, 745–54, Jun. 1974. [Online]. Available: http://adsabs.harvard.edu/cgi-bin/nph-bib_query?bibcode=1974AJ.....79..745L (cit. on p. 1.17).

richardson:72:bbi [74] W. H. Richardson, “Bayesian-based iterative method of image restoration,” *J. Opt. Soc. Am.*, vol. 62, no. 1, 55–9, Jan. 1972. DOI: 10.1364/JOSA.62.000055 (cit. on p. 1.17).

bickel:77 [75] P. J. Bickel and K. A. Doksum, *Mathematical statistics*. Oakland, CA: Holden-Day, 1977 (cit. on p. 1.18).

kay:93 [76] S. M. Kay, *Fundamentals of statistical signal processing: ESTIMATION theory*. New York: Prentice-Hall, 1993 (cit. on pp. 1.18, 1.20).

stark:86 [77] H. Stark and J. W. Woods, *Probability, random processes, and estimation theory for engineers*. Englewood Cliffs, NJ: Prentice-Hall, 1986 (cit. on p. 1.20).

golub:79:gcv [78] G. H. Golub, M. Heath, and G. Wahba, “Generalized cross-validation as a method for choosing a good ridge parameter,” *Technometrics*, vol. 21, no. 2, 215–23, May 1979. [Online]. Available: http://www.jstor.org/stable/1268518 (cit. on p. 1.21).

miller:70:lsm [79] K. Miller, “Least-squares methods for ill-posed problems with a prescribed bound,” *SIAM J. Math. Anal.*, vol. 1, no. 1, 52–70, Feb. 1970. DOI: 10.1137/0501006 (cit. on p. 1.22).

strang:99:tdc [80] G. Strang, “The discrete cosine transform,” *SIAM Review*, vol. 41, no. 1, 135–47, 1999. DOI: 10.1137/S0036144598336745 (cit. on p. 1.23).

twomey:63:otn [81] S. Twomey, “On the numerical solution of Fredholm integral equations of the first kind by the inversion of the linear system produced by quadrature,” *J. Assoc. Comput. Mach.*, vol. 10, no. 1, 97–101, Jan. 1963. DOI: 10.1145/321150.321157 (cit. on p. 1.24).

kunsch:94:rpf [82] H. Künsch, “Robust priors for smoothing and image restoration,” *Ann. Inst. Stat. Math.*, vol. 46, no. 1, 1–19, Mar. 1994. DOI: 10.1007/BF00773588 (cit. on p. 1.24).

hunt:73:tao [83] B. R. Hunt, “The application of constrained least squares estimation to image restoration by digital computers,” *IEEE Trans. Comp.*, vol. 22, no. 9, 805–12, Sep. 1973 (cit. on p. 1.24).

fessler:96:srp [84] J. A. Fessler and W. L. Rogers, “Spatial resolution properties of penalized-likelihood image reconstruction methods: Space-invariant tomographs,” *IEEE Trans. Im. Proc.*, vol. 5, no. 9, 1346–58, Sep. 1996. DOI: 10.1109/83.535846 (cit. on p. 1.25).

hebber:97:deo [85] E. Hebber, D. Oldenburg, T. Farnocombe, and A. Celler, “Direct estimation of dynamic parameters in SPECT tomography,” *IEEE Trans. Nuc. Sci.*, vol. 44, no. 6-2, 2425–30, Dec. 1997. DOI: 10.1109/23.656447 (cit. on p. 1.25).

elad:02:oto [86] M. Elad, “On the origin of the bilateral filter and ways to improve it,” *IEEE Trans. Im. Proc.*, vol. 11, no. 10, 1141–51, Oct. 2002. DOI: 10.1109/TIP.2002.801126 (cit. on p. 1.25).

fessler:00:edo [87] J. A. Fessler, H. Erdoνgan, and W. B. Wu, “Exact distribution of edge-preserving MAP estimators for linear signal models with Gaussian measurement noise,” *IEEE Trans. Im. Proc.*, vol. 9, no. 6, 1049–56, Jun. 2000. DOI: 10.1109/83.846247 (cit. on p. 1.25).

huber:81 [88] P. J. Huber, *Robust statistics*. New York: Wiley, 1981 (cit. on pp. 1.31, 1.33).

allain:02:oga [89] M. Allain, J. Idier, and Y. Goussard, “On global and local convergence of half-quadratic algorithms,” in *Proc. IEEE Intl. Conf. on Image Processing*, vol. 2, 2002, II–833-6. DOI: 10.1109/ICIP.2002.1040080 (cit. on p. 1.31).

li:98:cfs [90] S. Z. Li, “Close-form solution and parameter selection for convex minimization-based edge-preserving smoothing,” *IEEE Trans. Patt. Anal. Mach. Int.*, vol. 20, no. 9, 916–32, Sep. 1998. DOI: 10.1109/34.713359 (cit. on p. 1.32).

terzopoulos:86:roi [91] D. Terzopoulos, “Regularization of inverse visual problems involving discontinuities,” *IEEE Trans. Patt. Anal. Mach. Int.*, vol. 8, no. 4, 413–24, Jul. 1986. DOI: 10.1109/TPAMI.1986.4767807 (cit. on p. 1.32).

jeon:00:iru [92] W. Jeon and T. Yi, “Image restoration using dual adaptive regularization operators,” in *Proc. IEEE Intl. Conf. on Pattern Recognition*, vol. 3, 2000, 45–8 (cit. on p. 1.32).

park:00:nae [93] S. C. Park and M. G. Kang, "Noise-adaptive edge-preserving image restoration algorithm," *Optical Engineering*, vol. 39, no. 12, 3124–37, Dec. 2000 (cit. on p. 1.32).

perry:00:waf [94] S. W. Perry and L. Guan, "Weight assignment for adaptive image restoration by neural networks," *IEEE Trans on Neural Networks*, vol. 11, 156–70, 2000 (cit. on p. 1.32).

palmer:02:sai [95] A. Palmer, M. Razaz, and D. Mandic, "Spatially adaptive image restoration by neural network filtering," in *IEEE Proc. VII Brazilian Symposium on Neural Networks*, 2002, 184–9 (cit. on p. 1.32).

perry:02 [96] S. Perry, H. Wong, and L. Guan, *Adaptive image processing - A computational intelligence perspective*. CRC, 2002 (cit. on p. 1.32).

wu:04:rir [97] X. Wu, R. Wang, and C. Wang, "Regularized image restoration based on adaptively selecting parameter and operator," in *Proc. IEEE Intl. Conf. on Pattern Recognition*, vol. 3, 2004, 662–5. DOI: 10.1109/ICPR.2004.1334616 (cit. on p. 1.32).

geiger:91:pad [98] D. Geiger and F. Girosi, "Parallel and deterministic algorithms from MRF's: Surface reconstruction," *IEEE Trans. Patt. Anal. Mach. Int.*, vol. 13, no. 5, 401–12, May 1991. DOI: 10.1109/34.134040 (cit. on p. 1.36).

geman:92:cra [99] D. Geman and G. Reynolds, "Constrained restoration and the recovery of discontinuities," *IEEE Trans. Patt. Anal. Mach. Int.*, vol. 14, no. 3, 367–83, Mar. 1992. DOI: 10.1109/34.120331 (cit. on p. 1.36).

geman:95:nir [100] D. Geman and C. Yang, "Nonlinear image recovery with half-quadratic regularization," *IEEE Trans. Im. Proc.*, vol. 4, no. 7, 932–46, Jul. 1995. DOI: 10.1109/83.392335 (cit. on p. 1.36).

black:96:otu [101] M. J. Black and A. Rangarajan, "On the unification of line processes, outlier rejection, and robust statistics with applications in early vision," *Intl. J. Comp. Vision*, vol. 19, no. 1, 57–91, Jul. 1996. DOI: 10.1007/BF00131148 (cit. on p. 1.36).

raj:07:bpi [102] A. Raj, G. Singh, R. Zabih, B. Kressler, Y. Wang, N. Schuff, and M. Weiner, "Bayesian parallel imaging with edge-preserving priors," *Mag. Res. Med.*, vol. 57, no. 1, 8–21, Jan. 2007. DOI: 10.1002/mrm.21012 (cit. on p. 1.36).

geman:84:srg [103] S. Geman and D. Geman, "Stochastic relaxation, Gibbs distributions, and Bayesian restoration of images," *IEEE Trans. Patt. Anal. Mach. Int.*, vol. 6, no. 6, 721–41, Nov. 1984. DOI: 10.1109/TPAMI.1984.4767596 (cit. on p. 1.36).

derin:87:mas [104] H. Derin and H. Elliott, "Modeling and segmentation of noisy and textured images using Gibbs random fields," *IEEE Trans. Patt. Anal. Mach. Int.*, vol. 9, no. 1, 39–55, Jan. 1987. DOI: 10.1109/TPAMI.1987.4767871 (cit. on p. 1.36).

silverman:90:tso [105] B. W. Silverman, C. Jennison, J. Stander, and T. C. Brown, "The specification of edge penalties for regular and irregular pixel images," *IEEE Trans. Patt. Anal. Mach. Int.*, vol. 12, no. 10, 1017–24, Oct. 1990. DOI: 10.1109/34.58874 (cit. on p. 1.36).

jeng:91:cgm [106] F. C. Jeng and J. W. Woods, "Compound Gauss-Markov random fields for image estimation," *IEEE Trans. Sig. Proc.*, vol. 39, no. 3, 683–97, Mar. 1991. DOI: 10.1109/78.80887 (cit. on p. 1.36).

ambrosio:90:aof [107] L. Ambrosio and V. M. Tortorelli, "Approximation of functionals depending on jumps by elliptic functionals via Γ-convergence," *Comm. Pure Appl. Math.*, vol. 43, no. 8, 999–1036, Dec. 1990. DOI: 10.1002/cpa.3160430805 (cit. on p. 1.36).

johnson:91:iru [108] V. E. Johnson, W. H. Wong, X. Hu, and C. T. Chen, "Image restoration using Gibbs priors: Boundary modeling, treatment of blurring, and selection of hyperparameter," *IEEE Trans. Patt. Anal. Mach. Int.*, vol. 13, no. 5, 413–25, May 1991. DOI: 10.1109/34.134041 (cit. on p. 1.36).

johnson:93:aff [109] V. Johnson, "A framework for incorporating structural prior information into the estimation of medical images," in *Information Processing in Medical Im.* H. H. Barrett and A. F. Gmitro, Eds., Berlin: Springer Verlag, 1993, pp. 307–21 (cit. on p. 1.36).

johnson:94:amf [110] V. E. Johnson, "A model for segmentation and analysis of noisy images," *J. Am. Stat. Assoc.*, vol. 89, no. 425, 230–41, Mar. 1994 (cit. on p. 1.36).

bowsher:96:bra [111] J. E. Bowsher, V. E. Johnson, T. G. Turkington, R. J. Jaszczak, C. E. Floyd, and R. E. Coleman, "Bayesian reconstruction and use of anatomical a priori information for emission tomography," *IEEE Trans. Med. Imag.*, vol. 15, no. 5, 673–86, Oct. 1996. DOI: 10.1109/42.538945 (cit. on p. 1.36).

yu:98:ept [112] D. F. Yu and J. A. Fessler, "Edge-preserving tomographic reconstruction with nonlocal regularization," in *Proc. IEEE Intl. Conf. on Image Processing*, vol. 1, 1998, 29–33. DOI: 10.1109/ICIP.1998.723402 (cit. on p. 1.36).

yu:02:ept [113] ——, "Edge-preserving tomographic reconstruction with nonlocal regularization," *IEEE Trans. Med. Imag.*, vol. 21, no. 2, 159–73, Feb. 2002. DOI: 10.1109/42.993134 (cit. on p. 1.36).

ye:02:asr

[114] J. C. Ye, Y. Bresler, and P. Moulin, "A self-referencing level-set method for image reconstruction from sparse Fourier samples," *Intl. J. Comp. Vision*, vol. 50, no. 3, 253–70, Dec. 2002. DOI: 10.1023/A:1020822324006 (cit. on p. 1.36).

weisenseel:02:sbf

[115] R. A. Weisenseel, W. C. Karl, and R. C. Chan, "Shared-boundary fusion for estimation of noisy multi-modality atherosclerotic plaque imagery," in *Proc. IEEE Intl. Conf. on Image Processing*, vol. 3, 2002, 157–60. DOI: 10.1109/ICIP.2002.1038929 (cit. on p. 1.36).

frieden:72:rwm

[116] B. R. Frieden, "Restoring with maximum likelihood and maximum entropy," *J. Opt. Soc. Am.*, vol. 62, no. 4, 511–8, Apr. 1972. DOI: 10.1364/JOSA.62.000511 (cit. on p. 1.36).

gull:78:irf

[117] S. F. Gull and G. J. Daniell, "Image reconstruction from incomplete and noisy data," *Nature*, vol. 272, no. 5655, 686–90, Apr. 1978. DOI: 10.1038/272686a0 (cit. on p. 1.36).

minerbo:79:mam

[118] G. Minerbo, "MENT: A Maximum entropy algorithm for reconstructing a source from projection data," *Comp. Graphics and Im. Proc.*, vol. 10, no. 1, 48–68, May 1979. DOI: 10.1016/0146-664X(79)90034-0 (cit. on p. 1.36).

trussell:80:trb

[119] H. J. Trussell, "The relationship between image restoration by the maximum a posteriori method and a maximum entropy method," *IEEE Trans. Acoust. Sp. Sig. Proc.*, vol. 28, no. 12, 114–7, Feb. 1980. DOI: 10.1109/TASSP.1980.1163348 (cit. on p. 1.36).

jaynes:82:otr

[120] E. T. Jaynes, "On the rationale of maximum entropy methods," *Proc. IEEE*, vol. 70, no. 9, 939–52, Sep. 1982. DOI: 10.1109/PROC.1982.12425 (cit. on p. 1.36).

burch:83:irb

[121] S. F. Burch, S. F. Gull, and J. Skilling, "Image restoration by a powerful maximum entropy method," *Comp. Vision, Graphics, and Im. Proc.*, vol. 23, no. 2, 113–28, Aug. 1983. DOI: 10.1016/0734-189X(83)90108-1 (cit. on p. 1.36).

gull:84:mem

[122] S. F. Gull and J. Skilling, "Maximum entropy method in image processing," *IEE Proceedings: F Communications*, vol. 131, no. 6, 646–59, Oct. 1984 (cit. on p. 1.36).

skilling:84:mei

[123] J. Skilling and R. K. Bryan, "Maximum entropy image reconstruction: General algorithm," *Monthly Notices Royal Astron. Soc.*, vol. 211, no. 1, 111–24, Nov. 1984. [Online]. Available: http://adsabs.harvard.edu/cgi-bin/nph-bib_query?bibcode=1984MNRAS.211..111S (cit. on p. 1.36).

censor:87:obi

[124] Y. Censor and J. Segman, "On block-iterative entropy maximization," *Journal of Information and Optimization Sciences*, vol. 8, no. 3, 275–91, 1987 (cit. on p. 1.36).

csiszar:91:wls

[125] I. Csiszár, "Why least squares and maximum entropy? An axiomatic approach to inference for linear inverse problems," *Ann. Stat.*, vol. 19, no. 4, 2032–66, 1991. DOI: 10.1214/aos/1176348385 (cit. on p. 1.36).

myrheim:92:naf

[126] J. Myrheim and H. Rue, "New algorithms for maximum entropy image restoration," *Comp. Vision, Graphics, and Im. Proc.: Graph. Models Im. Proc.*, vol. 54, no. 3, 223–8, May 1992. DOI: 10.1016/1049-9652(92)90053-Z (cit. on p. 1.36).

eggermont:93:mer

[127] P. P. B. Eggermont, "Maximum entropy regularization for Fredholm integral equation of the first kind," *SIAM J. Math. Anal.*, vol. 24, no. 6, 1557–76, 1993. DOI: 10.1137/0524088 (cit. on p. 1.36).

lyon:97:ame

[128] R. G. Lyon, J. M. Hollis, and J. E. Dorband, "A maximum entropy method with a priori maximum likelihood constraints," *The Astrophysical Journal*, vol. 478, no. 2-1, 658–62, Apr. 1997. DOI: 10.1086/303837 (cit. on p. 1.36).

jannetta:04:mir

[129] A. Jannetta, J. C. Jackson, C. J. Kotre, I. P. Birch, K. J. Robson, and R. Padgett, "Mammographic image restoration using maximum entropy deconvolution," *Phys. Med. Biol.*, vol. 49, no. 21, 4997–5010, Nov. 2004. DOI: 10.1088/0031-9155/49/21/011 (cit. on p. 1.36).

papoulis:81:mea

[130] A. Papoulis, "Maximum entropy and spectral estimation: A review," *IEEE Trans. Acoust. Sp. Sig. Proc.*, vol. 29, no. 6, 1176–86, Dec. 1981. DOI: 10.1109/TASSP.1981.1163713 (cit. on p. 1.36).

gullberg:75:eat

[131] G. T. Gullberg, "Entropy and transverse section reconstruction," in *Information Processing in Scintigraphy: Proceedings of the IVth International Conference*, C. Raynaud and A. Todd-Pokropek, Eds., Orsay, France: Commissariat à l'Énergie Atomique, 1975, pp. 325–32 (cit. on p. 1.36).

gullberg:87:mer

[132] G. Gullberg and B. M. W. Tsui, "Maximum entropy reconstruction with constraints: Iterative algorithms for solving the primal and dual programs," in *Proc. Tenth Intl. Conf. on Information Processing in Medical Im.* C. N. de Graaf and M. A. Viergever, Eds., New York: Plenum Press, 1987, pp. 181–200 (cit. on p. 1.36).

mohammaddjafari:88:mei

[133] A. Mohammad-Djafari and G. Demoment, "Maximum entropy image reconstruction in X-ray and diffraction tomography," *IEEE Trans. Med. Imag.*, vol. 7, no. 4, 345–54, Dec. 1988. DOI: 10.1109/42.14518 (cit. on p. 1.36).

desmedt:93:aco

[134] P. Desmedt and I. Lemahieu, "A comparison of different prior laws for Bayesian image reconstruction in positron emission tomography," in *Maximum Entropy and Bayesian Methods*, A. Mohammad-Djafari and G. Demoments, Eds., Netherlands: Kluwer, 1993, pp. 393–8 (cit. on p. 1.36).

lalush:00:fme
[135] D. S. Lalush, E. C. Frey, and B. M. W. Tsui, “Fast maximum entropy approximation in SPECT using the RBI-MAP algorithm,” *IEEE Trans. Med. Imag.*, vol. 19, no. 4, 286–94, Apr. 2000. DOI: 10.1109/42.848180 (cit. on p. 1.36).

elfving:80:osm
[136] T. Elfving, “On some methods for entropy maximization and matrix scaling,” *Linear Algebra and its Applications*, vol. 34, 321–9, Dec. 1980. DOI: 10.1016/0024-3795(80)90171-8 (cit. on p. 1.36).

donoho:90:dtm
[137] D. L. Donoho, I. M. Johnstone, A. S. Stern, and J. Hoch, “Does the maximum entropy method improve sensitivity,” *Proc. Natl. Acad. Sci.*, vol. 87, no. 13, 5066–8, Jul. 1990. [Online]. Available: http://www.pnas.org/cgi/content/abstract/87/13/5066 (cit. on p. 1.36).

donoho:92:mea
[138] D. L. Donoho, I. M. Johnstone, J. C. Hoch, and A. S. Stern, “Maximum entropy and the nearly black object,” *J. Royal Stat. Soc. Ser. B*, vol. 54, no. 1, 41–81, 1992. [Online]. Available: http://www.jstor.org/stable/2345948 (cit. on p. 1.36).

jeffs:93:rob
[139] B. D. Jeffs and M. Gunsay, “Restoration of blurred star field images by maximally sparse optimization,” *IEEE Trans. Im. Proc.*, vol. 2, no. 2, 202–211, Apr. 1993. DOI: 10.1109/83.217223 (cit. on p. 1.37).

li:02:aai
[140] M. Li, H. Yang, and H. Kudo, “An accurate iterative reconstruction algorithm for sparse objects: Application to 3D blood vessel reconstruction from a limited number of projections,” *Phys. Med. Biol.*, vol. 47, no. 15, 2599–610, Aug. 2002. DOI: 10.1088/0031-9155/47/15/303 (cit. on p. 1.37).

duffin:52:aco
[141] R. J. Duffin and A. C. Schaeffer, “A class of nonharmonic Fourier series,” *Tams*, vol. 72, no. 2, 341–66, Mar. 1952. [Online]. Available: http://www.jstor.org/stable/1990760 (cit. on p. 1.37).

aharon:06:ksa
[142] M. Aharon, M. Elad, and A. Bruckstein, “K-SVD: an algorithm for designing overcomplete dictionaries for sparse representation,” *IEEE Trans. Sig. Proc.*, vol. 54, no. 11, 4311–22, Nov. 2006. DOI: 10.1109/TSP.2006.881199 (cit. on p. 1.37).

rubinstein:10:dfs
[143] R. Rubinstein, A. M. Bruckstein, and M. Elad, “Dictionaries for sparse representation modeling,” *Proc. IEEE*, vol. 98, no. 6, 1045–1057, Jun. 2010. DOI: 10.1109/JPROC.2010.2040551 (cit. on p. 1.37).

smith:13:idl
[144] L. N. Smith and M. Elad, “Improving dictionary learning: Multiple dictionary updates and coefficient reuse,” *IEEE Signal Proc. Letters*, vol. 20, no. 1, 79–82, Jan. 2013. DOI: 10.1109/LSP.2012.2229976 (cit. on p. 1.37).

ravishankar::eso
[145] S. Ravishankar, R. R. Nadakuditi, and J. A. Fessler, “Efficient sum of outer products dictionary learning (SOUP-DIL) - The ℓ_0 method,” *IEEE Trans. Sig. Proc.*, 2015, Submitted. (cit. on p. 1.37).

blumensath:08:itf
[146] T. Blumensath and M. E. Davies, “Iterative thresholding for sparse approximations,” *J. Fourier Anal. and Appl.*, vol. 14, no. 5, 629–54, 2008. DOI: 10.1007/s00041-008-9035-z (cit. on p. 1.37).

bruckstein:09:fss
[147] A. Bruckstein, D. Donoho, and M. Elad, “From sparse solutions of systems of equations to sparse modeling of signals and images,” *SIAM Review*, vol. 51, no. 1, 34–81, 2009. DOI: 10.1137/060657704 (cit. on p. 1.37).

sidky:08:iri
[148] E. Y. Sidky and X. Pan, “Image reconstruction in circular cone-beam computed tomography by constrained, total-variation minimization,” *Phys. Med. Biol.*, vol. 53, no. 17, 4777–808, Sep. 2008. DOI: 10.1088/0031-9155/53/17/021 (cit. on p. 1.37).

yukawa:12:ecl
[149] M. Yukawa and S. Amari, “ℓ_p-constrained least squares $(0 < p < 1)$ and its critical path,” in *Intl. Symp. on Information Theory*, 2012, 2221–5. DOI: 10.1109/ISIT.2012.6283848 (cit. on p. 1.37).

nikolova:13:dot
[150] M. Nikolova, “Description of the minimizers of least squares regularized with ℓ_0-norm. uniqueness of the global minimizer,” *SIAM J. Imaging Sci.*, vol. 6, no. 2, 904–37, 2013. DOI: 10.1137/11085476X (cit. on p. 1.37).

elad:07:avs
[151] M. Elad, P. Milanfar, and R. Rubinstein, “Analysis versus synthesis in signal priors,” *Inverse Prob.*, vol. 23, no. 3, 947–68, Jun. 2007. DOI: 10.1088/0266-5611/23/3/007 (cit. on p. 1.38).

tibshirani:96:rsa
[152] R. Tibshirani, “Regression shrinkage and selection via the LASSO,” *J. Royal Stat. Soc. Ser. B*, vol. 58, no. 1, 267–88, 1996. [Online]. Available: http://www.jstor.org/stable/2346178 (cit. on p. 1.38).

osborne:00:otl
[153] M. Osborne, B. Presnell, and B. Turlach, “On the LASSO and its dual,” *J. Computational and Graphical Stat.*, vol. 9, no. 2, 319–37, Jun. 2000. [Online]. Available: http://www.jstor.org/stable/1390657 (cit. on p. 1.38).

osborne:00:ana
[154] ——, “A new approach to variable selection in least squares problems,” *IMA J. Numerical Analysis*, vol. 20, no. 3, 389–404, Jul. 2000. DOI: 10.1093/imanum/20.3.389 (cit. on p. 1.38).

efron:04:lar
[155] B. Efron, T. Hastie, I. Johnstone, and R. Tibshirani, “Least angle regression,” *Ann. Stat.*, vol. 32, no. 2, 407–99, 2004. DOI: 10.1214/009053604000000067 (cit. on p. 1.38).

gasso:08:snc
[156] G. Gasso, A. Rakotomamonjy, and S. Canu, “Solving non-convex lasso type problems with DC programming,” in *Proc. IEEE Wkshp. Machine Learning for Signal Proc.*, 2008, 450–5. DOI: 10.1109/MLSP.2008.4685522 (cit. on p. 1.38).

wu:08:cda [157] T. T. Wu and K. Lange, "Coordinate descent algorithms for lasso penalized regression," *Annapplstat*, vol. 2, no. 1, 224–44, 2008. DOI: 10.1137/110840054 (cit. on p. 1.38).

donoho:93:svs [158] D. Donoho, "Superresolution via sparsity constraints," *SIAM J. Math. Anal.*, vol. 23, no. 5, 1309–31, 1993. DOI: 10.1137/0523074 (cit. on p. 1.38).

harikumar:96:ana [159] G. Harikumar and Y. Bresler, "A new algorithm for computing sparse solutions to linear inverse problems," in *Proc. IEEE Conf. Acoust. Speech Sig. Proc.*, vol. 3, 1996, 1331–4. DOI: 10.1109/ICASSP.1996.543672 (cit. on p. 1.38).

gorodnitsky:97:ssr [160] I. F. Gorodnitsky and B. D. Rao, "Sparse signal reconstruction from limited data using FOCUSS: a re-weighted minimum norm algorithm," *IEEE Trans. Sig. Proc.*, vol. 45, no. 3, 600–16, Mar. 1997. DOI: 10.1109/78.558475 (cit. on p. 1.38).

donoho:06:fml [161] D. Donoho, "For most large underdetermined systems of linear equations, the minimal ell-1 norm solution is also the sparsest solution," *Comm. Pure Appl. Math.*, vol. 59, no. 6, 797–829, Jun. 2006. DOI: 10.1002/cpa.20132 (cit. on p. 1.38).

candes:06:rup [162] E. J. Candès, J. Romberg, and T. Tao, "Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information," *IEEE Trans. Info. Theory*, vol. 52, no. 2, 489–509, Feb. 2006. DOI: 10.1109/TIT.2005.862083 (cit. on p. 1.38).

donoho:06:cs [163] D. L. Donoho, "Compressed sensing," *IEEE Trans. Info. Theory*, vol. 52, no. 4, 1289–1306, Apr. 2006. DOI: 10.1109/TIT.2006.871582 (cit. on p. 1.38).

donoho:06:sro [164] D. L. Donoho, M. Elad, and V. N. Temlyakov, "Stable recovery of sparse overcomplete representations in the presence of noise," *IEEE Trans. Info. Theory*, vol. 52, no. 1, 6–18, Jan. 2006. DOI: 10.1109/TIT.2005.860430 (cit. on p. 1.38).

figueiredo:07:gpf [165] M. Figueiredo, R. Nowak, and S. J. Wright, "Gradient projection for sparse reconstruction: Application to compressed sensing and other inverse problems," *IEEE J. Sel. Top. Sig. Proc.*, vol. 1, no. 4, 586–97, Dec. 2007. DOI: 10.1109/JSTSP.2007.910281. [Online]. Available: http://www.lx.it.pt/~mtf/GPSR/ (cit. on p. 1.38).

candes:07:sai [166] E. Candès and J. Romberg, "Sparsity and incoherence in compressive sampling," *Inverse Prob.*, vol. 23, no. 3, 969–86, Jun. 2007. DOI: 10.1088/0266-5611/23/3/008 (cit. on p. 1.38).

baraniuk:07:cs [167] R. G. Baraniuk, "Compressive sensing," *IEEE Sig. Proc. Mag.*, vol. 24, no. 4, 118–21, 2007. DOI: 10.1109/MSP.2007.4286571 (cit. on p. 1.38).

tsai:84:mfi [168] R. Y. Tsai and T. S. Huang, *Multi-frame image restoration and registration*, Adv. Comput. Vis. Image Process.: Image Reconstruction from Incomplete Observations vol 1 no 2 317-39 JAI Press, Greenwhich CT 1984, 1984 (cit. on p. 1.38).

irani:93:maf [169] M. Irani and S. Peleg, "Motion analysis for image enhancement: Resolution, occlusion, and transparency," *J. Visual Comm. Im. Rep.*, vol. 4, no. 4, 324–35, Dec. 1993. DOI: 10.1006/jvci.1993.1030 (cit. on p. 1.38).

eren:97:rob [170] P. E. Eren, M. I. Sezan, and A. M. Tekalp, "Robust, object-based high-resolution image reconstruction from low-resolution video," *IEEE Trans. Im. Proc.*, vol. 6, no. 10, 1446–51, Oct. 1997. DOI: 10.1109/83.624970 (cit. on p. 1.38).

elad:97:roa [171] M. Elad and A. Feuer, "Restoration of a single superresolution image from several blurred, noisy, and undersampled measured images," *IEEE Trans. Im. Proc.*, vol. 6, no. 12, 1646–58, Dec. 1997. DOI: 10.1109/83.650118 (cit. on p. 1.38).

hardie:97:jmr [172] R. C. Hardie, K. J. Barnard, and E. E. Armstrong, "Joint MAP registration and high-resolution image estimation using a sequence of undersampled images," *IEEE Trans. Im. Proc.*, vol. 6, no. 12, 1621–33, Dec. 1997. DOI: 10.1109/83.650116 (cit. on p. 1.38).

elad:99:srr [173] M. Elad and A. Feuer, "Super-resolution reconstruction of image sequences," *IEEE Trans. Patt. Anal. Mach. Int.*, vol. 21, no. 9, 817–34, Sep. 1999. DOI: 10.1109/34.790425 (cit. on p. 1.38).

elad:01:afs [174] M. Elad and Y. Hel-Or, "A fast super-resolution reconstruction algorithm for pure translational motion and common space-invariant blur," *IEEE Trans. Im. Proc.*, vol. 10, no. 8, 1187–93, Aug. 2001. DOI: 10.1109/83.935034 (cit. on p. 1.38).

nguyen:01:ace [175] N. Nguyen, P. Milanfar, and G. Golub, "A computationally efficient superresolution image reconstruction algorithm," *IEEE Trans. Im. Proc.*, vol. 10, no. 4, 573–83, Apr. 2001. DOI: 10.1109/83.913592 (cit. on p. 1.38).

nguyen:01:egc [176] ——, "Efficient generalized cross-validation with applications to parametric image restoration and resolution enhancement," *IEEE Trans. Im. Proc.*, vol. 10, no. 9, 1299–308, Sep. 2001. DOI: 10.1109/83.941854 (cit. on p. 1.38).

patti:01:arf
[177] A. J. Patti and Y. Altunbasak, "Artifact reduction for set theoretic super resolution image reconstruction with edge adaptive constraints and higher-order interpolants," *IEEE Trans. Im. Proc.*, vol. 10, no. 1, 179–86, Jan. 2001. DOI: 10.1109/83.892456 (cit. on p. 1.38).

lee:03:rah
[178] E. S. Lee and M. G. Kang, "Regularized adaptive high-resolution image reconstruction considering inaccurate subpixel registration," *IEEE Trans. Im. Proc.*, vol. 12, no. 7, 826–837, Jul. 2003. DOI: 10.1109/TIP.2003.811488 (cit. on p. 1.38).

park:03:sri
[179] S. C. Park, M. K. Park, and M. G. Kang, "Super-resolution image reconstruction: A technical overview," *IEEE Sig. Proc. Mag.*, vol. 20, no. 3, 21–36, May 2003. DOI: 10.1109/MSP.2003.1203207 (cit. on p. 1.38).

farsiu:04:far
[180] S. Farsiu, M. D. Robinson, M. Elad, and P. Milanfar, "Fast and robust multiframe super resolution," *IEEE Trans. Im. Proc.*, vol. 13, no. 10, 1327–44, Oct. 2004. DOI: 10.1109/TIP.2004.834669 (cit. on p. 1.38).

chung:06:nmf
[181] J. Chung, E. Haber, and J. Nagy, "Numerical methods for coupled super-resolution," *Inverse Prob.*, vol. 22, no. 4, 1261–72, Aug. 2006. DOI: 10.1088/0266-5611/22/4/009 (cit. on p. 1.38).

woods:06:smf
[182] N. A. Woods, N. P. Galatsanos, and A. K. Katsaggelos, "Stochastic methods for joint registration, restoration, and interpolation of multiple undersampled images," *IEEE Trans. Im. Proc.*, vol. 15, no. 1, 201–13, Jan. 2006. DOI: 10.1109/TIP.2005.860355 (cit. on p. 1.38).

chantas:07:srb
[183] G. K. Chantas, N. P. Galatsanos, and N. A. Woods, "Super-resolution based on fast registration and maximum a posteriori reconstruction," *IEEE Trans. Im. Proc.*, vol. 16, no. 7, 1821–30, Jul. 2007. DOI: 10.1109/TIP.2007.896664 (cit. on p. 1.38).

fransens:07:ofb
[184] R. Fransens, C. Strecha, and L. Van Gool, "Optical flow based super-resolution: A probabilistic approach," *Comp. Vision & Im. Understanding*, vol. 106, no. 1, 106–15, Apr. 2007. DOI: 10.1016/j.cviu.2005.09.011 (cit. on p. 1.38).

farsiu:04:aac
[185] S. Farsiu, D. Robinson, M. Elad, and P. Milanfar, "Advances and challenges in super-resolution," *Intl. J. Imaging Sys. and Tech.*, vol. 14, no. 2, 47–57, 2004. DOI: 10.1002/ima.20007 (cit. on p. 1.38).

milanfar:10
[186] P. Milanfar and Editor, *Super-resolution imaging*. CRC, 2010. [Online]. Available: http://www.amazon.com/gp/product/1439819300/ (cit. on p. 1.38).

liang:92:crm
[187] Z. P. Liang, F. E. Boada, R. T. Constable, E. M. Haacke, P. C. Lauterbur, and M. R. Smith, "Constrained reconstruction methods in MR imaging," *Reviews of Magnetic Resonance in Medicine*, vol. 4, 67–185, 1992 (cit. on p. 1.38).

verveer:98:irf
[188] P. J. Verveer and T. M. Jovin, "Improved restoration from multiple images of a single object: Application to fluorescence microscopy," *Appl. Optics*, vol. 37, no. 26, 6240–6, Sep. 1998. DOI: 10.1364/AO.37.006240 (cit. on p. 1.38).

peled:01:sim
[189] S. Peled and Y. Yeshurun, "Superresolution in MRI: Application to human white matter fiber tract visualization by diffusion tensor imaging," *Mag. Res. Med.*, vol. 45, no. 1, 29–35, Jan. 2001. DOI: 10.1002/1522-2594(200101)45:1<29::AID-MRM1005>3.0.CO;2-Z (cit. on p. 1.38).

greenspan:02:mis
[190] H. Greenspan, G. Oz, N. Kiryati, and S. Peled, "MRI inter-slice reconstruction using super-resolution," *Mag. Res. Im.*, vol. 20, no. 5, 437–46, Jun. 2002. DOI: 10.1016/S0730-725X(02)00511-8 (cit. on p. 1.38).

peled:02:sim
[191] S. Peled and Y. Yeshurun, "Superresolution in MRI - perhaps sometimes," *Mag. Res. Med.*, vol. 48, no. 2, p. 409, Aug. 2002. DOI: 10.1002/mrm.10237 (cit. on p. 1.38).

scheffler:02:sim
[192] K. Scheffler, "Superresolution in MRI?" *Mag. Res. Med.*, vol. 48, no. 2, p. 408, Aug. 2002. DOI: 10.1002/mrm.10203 (cit. on p. 1.38).

desjardins:03:sfc
[193] B. Desjardins and T. L. Chenevert, "Superresolution from coplanar image sequences in MRI," in *Proc. Intl. Soc. Mag. Res. Med.*, 2003, p. 1074. [Online]. Available: http://cds.ismrm.org/ismrm-2003/1074.pdf (cit. on p. 1.38).

peeters:04:tuo
[194] R. R. Peeters, P. Kornprobst, M. Nikolova, S. Sunaert, T. Vieville, G. Malandain, R. Deriche, O. Faugeras, M. Ng, and P. V. Hecke, "The use of super-resolution techniques to reduce slice thickness in functional MRI," *Intl. J. Imaging Sys. and Tech.*, vol. 14, no. 3, 131–8, 2004. DOI: 10.1002/ima.20016 (cit. on p. 1.38).

kennedy:07:iif
[195] J. A. Kennedy, O. Israel, A. Frenkel, R. Bar-Shalom, and H. Azhari, "Improved image fusion in PET/CT using hybrid image reconstruction and super-resolution," *Intl. J. Biomedical Im.*, vol. 2007, p. 46 846, 2007. DOI: 10.1155/2007/46846 (cit. on p. 1.38).

greenspan:09:sri
[196] H. Greenspan, "Super-resolution in medical imaging," *The Computer Journal*, vol. 52, no. 1, 43–63, 2009. DOI: 10.1093/comjnl/bxm075 (cit. on p. 1.38).

kulkarni:99:eot
[197] D. Kulkarni, D. Schmidt, and S.-K. Tsui, "Eigenvalues of tridiagonal pseudo-Toeplitz matrices," *Linear Algebra and its Applications*, vol. 297, no. 1-3, 63–80, Aug. 1999. DOI: 10.1016/S0024-3795(99)00114-7.

chan:96:cgm

[198] R. H. Chan and M. K. Ng, "Conjugate gradient methods for Toeplitz systems," *SIAM Review*, vol. 38, no. 3, 427–82, Sep. 1996. DOI: 10.1137/S0036144594276474. [Online]. Available: http://www.jstor.org/stable/2132496 (cit. on p. 1.39).

serra:98:ote

[199] S. Serra, "On the extreme eigenvalues of hermitian (Block) Toeplitz matrices," *Linear Algebra and its Applications*, vol. 270, 109–29, Feb. 1998. DOI: '10.1016/S0024-3795(98)80014-1' (cit. on p. 1.39).

tyrtyshnikov:96:aua

[200] E. E. Tyrtyshnikov, "A unifying approach to some old and new theorems on distribution and clustering," *Linear Algebra and its Applications*, vol. 232, no. 1, 1–43, Jan. 1996. DOI: 10.1016/0024-3795(94)00025-5 (cit. on p. 1.39).

gray:72:ota

[201] R. Gray, "On the asymptotic eigenvalue distribution of Toeplitz matrices," *IEEE Trans. Info. Theory*, vol. 18, no. 6, 725–30, Nov. 1972. DOI: 10.1109/TIT.1972.1054924 (cit. on p. 1.39).

grenander:84

[202] U. Grenander and G. Szegö, *Toeplitz forms and their applications*. New York: Chelsea, 1984 (cit. on p. 1.39).

dembo:88:bot

[203] A. Dembo, "Bounds on the extreme eigenvalues of positive-definite Toeplitz matrices," *IEEE Trans. Info. Theory*, vol. 34, no. 2, 352–5, Mar. 1988. DOI: 10.1109/18.2651 (cit. on p. 1.39).

voois:96:ato

[204] P. A. Voois, "A theorem on the asymptotic eigenvalue distribution of Toeplitz-block-Toeplitz matrices," *IEEE Trans. Sig. Proc.*, vol. 44, no. 7, 1837–41, Jul. 1996. DOI: 10.1109/78.510633 (cit. on p. 1.39).

chaux:07:avf

[205] C. Chaux, P. L. Combettes, J.-C. Pesquet, and V. R. Wajs, "A variational formulation for frame-based inverse problems," *Inverse Prob.*, vol. 23, no. 4, 1495–518, Aug. 2007. DOI: 10.1088/0266-5611/23/4/008 (cit. on p. 1.40).

antoniadis:01:raw

[206] A. Antoniadis and J. Fan, "Regularization and wavelet approximations," *J. Am. Stat. Assoc.*, vol. 96, no. 455, 939–55, Sep. 2001. [Online]. Available: http://www.jstor.org/stable/2670237 (cit. on p. 1.40).

weaver:91:fnf

[207] J. B. Weaver, Y. Xu, D. M. Healy Jr, and L. D. Cromwell, "Filtering noise from images with wavelet transforms," *Mag. Res. Med.*, vol. 21, no. 2, 288–95, Oct. 1991. DOI: 10.1002/mrm.1910210213 (cit. on p. 1.40).

gao:97:wwf

[208] H.-Y. Gao and A. G. Bruce, "WaveShrink with firm shrinkage," *Statistica Sinica*, vol. 7, no. 4, 855–74, Oct. 1997. [Online]. Available: http://www3.stat.sinica.edu.tw/statistica/j7n4/j7n43/j7n43.htm (cit. on p. 1.40).

unser:91:rrf

[209] M. Unser, A. Aldroubi, and M. Eden, "Recursive regularization filters: Design, properties, and applications," *IEEE Trans. Patt. Anal. Mach. Int.*, vol. 13, no. 3, 272–7, Mar. 1991. DOI: 10.1109/34.75514 (cit. on p. 1.40).

nuyts:03:apl

[210] J. Nuyts and J. A. Fessler, "A penalized-likelihood image reconstruction method for emission tomography, compared to post-smoothed maximum-likelihood with matched spatial resolution," *IEEE Trans. Med. Imag.*, vol. 22, no. 9, 1042–52, Sep. 2003. DOI: 10.1109/TMI.2003.816960 (cit. on p. 1.40).

kim:09:etf

[211] S. Kim, K. Koh, S. Boyd, and D. Gorinevsky, "ℓ_1 trend filtering," *SIAM Review*, vol. 51, no. 2, 339–60, Jun. 2009. DOI: 10.1137/070690274 (cit. on p. 1.40).

bracewell:00

[212] R. Bracewell, *The Fourier transform and its applications*, 3rd ed. New York: McGraw-Hill, 2000.

stark:92:pbi

[213] H. Stark and E. T. Olsen, "Projection-based image restoration," *J. Opt. Soc. Am. A*, vol. 9, no. 11, 1914–9, Nov. 1992. DOI: 10.1364/JOSAA.9.001914 (cit. on p. 1.41).

besag:86:ots

[214] J. Besag, "On the statistical analysis of dirty pictures," *J. Royal Stat. Soc. Ser. B*, vol. 48, no. 3, 259–302, 1986. [Online]. Available: http://www.jstor.org/stable/2345426 (cit. on p. 1.43).