CASCADIC MULTILEVEL METHODS FOR FAST NONSYMMETRIC BLUR- AND NOISE-REMOVAL

S. MORIGI*, L. REICHEL†, AND F. SGALLARI‡

Dedicated to Richard S. Varga on the occasion of his 80th birthday.

Abstract. Image deblurring is a discrete ill-posed problem. This paper discusses cascadic multilevel methods designed for the restoration of images that have been contaminated by nonsymmetric blur and noise. Prolongation is carried out by nonlinear edge-preserving and noise-reducing operators, while restrictions are determined by weighted local least-squares approximation. The restoration problem is on each level solved by an iterative method, with the number of iterations determined by the discrepancy principle. The performance of several iterative methods is compared. Computed examples demonstrate the effectiveness of the image restoration methods proposed. The discrepancy principle requires that an estimate of the norm of the noise in the contaminated image be available. We illustrate how such an estimate can be computed with the aid of the nonlinear Perona-Malik diffusion equation.

Key words. ill-posed problem, deblurring, multilevel method, regularizing iterative method, edge-preserving prolongation

1. Introduction. The restoration of images that have been contaminated by blur and noise continues to receive considerable attention. The blurring may be caused by object motion, calibration error of imaging devices, and random fluctuation of the medium, e.g., the atmosphere. The noise typically stems from the measurement equipment or transmission errors. We are interested in restoring images that have been contaminated by both blur and noise. In particular, we would like to recover edges accurately.

This paper considers the restoration of two-dimensional gray-scale images. These images can be represented by a real-valued function defined on a rectangular region $\Omega \subset \mathbb{R}^2$ or by the discretization of such a function. Let the function f^δ represent the available observed blur- and noise-contaminated image and let the function $\hat{u}$ represent the associated (unavailable) blur- and noise-free image that we would like to recover. These functions are assumed to be related by the degradation model

$$f^\delta(x) = \int_\Omega h(x,y)\hat{u}(y)dy + \eta^\delta(x), \qquad x \in \Omega, \tag{1.1}$$

where η^δ represents additive noise in the available data f^δ; see, e.g., Chan and Shen [8] for a discussion on image models.

In many applications, the integral is a convolution, i.e., $h(x,y) = k(x-y)$ for some function k. For instance, Gaussian kernels $k(x) = c_1 \exp(-c_2 x^2)$ with suitable positive constants c_1 and c_2 often are used to model atmospheric blur. The kernels h or k commonly are referred to as point spread functions (PSFs). In applications of interest, the kernel is smooth or piecewise smooth and, therefore, the integral operator is compact. It follows that the singular values of the integral operator cluster at the origin, leading to that the integral operator has no bounded inverse.

*Department of Mathematics-CIRAM, University of Bologna, Piazza Porta S. Donato 5, 40126 Bologna, Italy. E-mail: morigi@dm.unibo.it.

†Department of Mathematical Sciences, Kent State University, Kent, OH 44242, USA. E-mail: reichel@math.kent.edu.

‡Department of Mathematics-CIRAM, University of Bologna, Via Saragozza 8, 40123 Bologna, Italy. E-mail: sgallari@dm.unibo.it.

We would like to determine an accurate approximation of $\hat{u}$ when the observed image f^δ and the kernel h are known. Straightforward solution of the linear model

$$\int_\Omega h(x,y)u(y)dy = f^\delta(x), \qquad x \in \Omega, \tag{1.2}$$

for u typically does not provide a meaningful approximation of the desired noise- and blur-free image $\hat{u}$ due to the noise η^δ in the right-hand side f^δ and the fact that the integral operator does not have a bounded inverse. The latter property makes the task of solving (1.2) an ill-posed problem; see, e.g., Engl et al. [11] for analyses of ill-posed problems and discussions on numerical methods for their solution.

In order to be able to determine a meaningful approximation of $\hat{u}$, we first replace (1.2) by a nearby problem, whose solution is less sensitive to perturbations in the data f^δ, and then solve the new problem so obtained. This replacement is commonly referred to as regularization. The possibly most popular regularization approach is due to Tikhonov. The following form of Tikhonov regularization is well suited for image restoration,

$$\min_u \left\{ \int_\Omega \left(\frac{1}{2} \left(\int_\Omega h(x,y)u(y)dy - f^\delta(x) \right)^2 + \alpha\, R(u(x)) \right) dx \right\}, \tag{1.3}$$

where $\alpha > 0$ is a regularization parameter and

$$R(u) = \psi(|\nabla u|^2) \tag{1.4}$$

is a regularization operator. Here ψ is a differentiable monotonically increasing function and ∇u denotes the gradient of u; see, e.g., Rudin et al. [27] and Welk et al. [31] for discussions on this kind of regularization operators.

The Euler-Lagrange equation associated with (1.3), supplied with a gradient descent which yields a minimizer of (1.3) as $t \to \infty$, is given by

$$\frac{\partial u}{\partial t}(t,z) = -\int_\Omega h(x,z) \left(\int_\Omega h(x,y)u(t,y)dy - f^\delta(x) \right) dx + \alpha\, D(u(t,z)) \tag{1.5}$$

for $z \in \Omega$ and $t \geq 0$. We use the initial function $u(0,z) = f^\delta(z)$, $z \in \Omega$. Let $\partial\Omega$ denote the boundary of Ω. The derivation of (1.5) is based on the Dirichlet boundary condition $u(0,z) = 0$, $z \in \partial\Omega$. Other boundary conditions, such as Neumann conditions, also can be used and may in some situations give more accurate restorations; see, e.g., [22, 31] for discussions.

We also refer to D as a regularization operator. Image restoration methods based on the Euler-Lagrange equation require that the regularization operator D, as well as values of the regularization parameter α and a suitable finite interval of integration $[0,T]$ be chosen. The determination of suitable values of α and T generally is not straightforward.

From (1.4), we obtain

$$D(u) = \mathrm{div}(g(|\nabla u|^2)\nabla u), \qquad g(t) = d\psi(t)/dt. \tag{1.6}$$

The function g is referred to as the diffusivity. For instance, the choice $\psi(t) = t$ gives $g(t) = 1$ and $D(u) = \Delta u$, where Δ denotes the Laplacian. This regularization operator typically yields over-smoothed restored images. Total variation-type and

Perona-Malik regularization are designed to reduce over-smoothing. Perona-Malik regularization is obtained by choosing the diffusivity

$$g(s) = \frac{1}{1 + s/\rho}, \tag{1.7}$$

where ρ is a positive constant; see [24]. Total variation-type regularization is commented on below.

The present paper extends the multilevel image restoration methods described in [20] in several ways. We allow nonsymmetric PSFs. This requires the use of different iterative methods than in [20], and a careful comparison of the performance of these methods is provided. The restriction operators are defined by solving local weighted least-squares problems. The purpose of the weights is to avoid smearing of edges. Computed examples show that restriction by solving local least-squares problems gives more faithful restorations than restriction by local averaging. The latter approach to restriction is applied in [20]. The prolongation operators in our multilevel method are defined by piecewise linear prolongation followed by integration of the nonlinear Perona-Malik diffusion equation, determined by (1.6) with g given by (1.7), for a few time steps. These nonlinear prolongation operators are compared to nonlinear prolongation operators defined via total variation-type regularization in [20] and found to give restorations of somewhat higher accuracy. For theoretical purpose, we also discuss prolongation operators based on the linearized Perona-Malik diffusion equation. These linear prolongation operators yield linear image restoration methods, which we show to be a regularization methods in a well-defined sense. The quality of the restored images obtained with the linearized prolongation operators is almost as high as when nonlinear prolongation operators, based on the nonlinear Perona-Malik diffusion equation, are used. Finally, we explore the possibility of estimating the noise-level in the available corrupted image by integrating the Perona-Malik diffusion equation a few time steps.

We conclude this section with discussions on the discretizations of the linear integral equation (1.2) and of the Euler-Lagrange equation (1.5), and comment on the computational effort required for their solution. Discretization of (1.2) yields the linear system of equations

$$Au = b^\delta, \qquad A \in \mathbb{R}^{n\times n}, \qquad u, b^\delta \in \mathbb{R}^n, \tag{1.8}$$

where A is a discrete blurring operator and b^δ represents the available blur- and noise-contaminated image. Throughout this paper, the matrix A is assumed to be nonsymmetric, because otherwise we can employ the methods discussed in [20]. Since the integral operator in (1.2) has many singular values of different orders of magnitude close to the origin, so does the matrix A. The matrix therefore is of ill-determined rank and numerically singular. Linear systems of equations with a matrix of this kind often are referred to as linear discrete ill-posed problems. In image restoration applications, the right-hand side b^δ of (1.8), rather than the right-hand side f^δ of (1.2), is available. Each entry of b^δ represents a pixel value. Moreover, a measured discrete PSF, from which A is constructed, rather than the PSF h, may be known. The matrix A typically is very large and therefore cannot be factored.

Let $\hat{u}$ also denote a discrete approximation of the desired continuous solution of (1.1). Then the vector

$$\hat{b} = A\hat{u} \tag{1.9}$$

represents a blurred, but noise-free, image. Thus, (1.9) is a consistent linear system of equations for $\hat{u}$. Our task is to determine an approximate solution of (1.8) that accurately approximates $\hat{u}$. Note that due to the error

$$e^{\delta} = b^{\delta} - \hat{b} \tag{1.10}$$

in b^{δ} and the severe ill-conditioning of A, the minimal-norm least-squares solution of (1.8) generally is not a useful approximation of $\hat{u}$. We refer to the error e^{δ} as "noise." Note that in image restoration applications, it suffices to determine the discrete image $\hat{u}$.

A simple approach to determining an approximation of the desired image $\hat{u}$ is to apply a few steps of an iterative method to the linear system of equations (1.8) with initial iterate zero. Analyses of many classical iterative methods are provided by Varga [29]. Most iterative methods applied in the present paper are described by Saad [28]. Specifically, we consider the conjugate gradient method applied to the normal equations (LSQR), the generalized minimal residual (GMRES) iterative method, as well as the closely related range restricted GMRES (RRGMRES) iterative method. The iteration number may be considered a discrete regularization parameter. It is important not to carry out too many iterations in order to avoid severe error propagation. This approach of determining a restored image is referred to as regularization by truncated iteration; see [2, 4, 5, 6, 11, 14, 18, 19] for discussions. For many image restoration problems, LSQR and RRGMRES, and for some problems also GMRES, yield reasonable results within only a few iterations. The application of these methods therefore typically is quite inexpensive. However, due to cut-off of high frequencies, all of these iterative methods may introduce artifacts, such as "ringing" and fail to recover edges accurately; see [7, Example 4.3] for an illustrative example with LSQR and GMRES. Computed examples of Section 5 show that this shortcoming can be reduced by applying these iterative schemes in cascadic multilevel methods with suitable restriction and prolongation operators.

We turn to the discretization of the Euler-Lagrange equation (1.5) with D determined by (1.6) and (1.7). Space discretization can be carried out in several ways. For instance, finite difference or finite volume discretizations in space yield

$$\frac{du}{dt} = (\alpha\, L(u) - A^T A)u + A^T b^{\delta}, \tag{1.11}$$

where $L(u)u$ is a discretization of the regularization operator $D(u)$ in the right-hand side of (1.5). Thus, $L(u)$ is a discrete nonlinear diffusion operator. Explicit time stepping methods for the integration of (1.11) are expensive, because due to the CFL stability condition very many time steps have to be carried out; see [31] and [20, Example 4.2]. Semi-implicit time stepping methods allow longer time steps than explicit integration methods, but are not unconditionally stable due to the operator $-A^T A$. Each time step requires the solution of a large linear systems of equations. Therefore, semi-implicit integration methods also are expensive. We conclude that nonlinear models based on (1.5)-(1.7) and (1.11) can provide denoising and deblurring of good quality; however, the solution of (1.11) by explicit or semi-implicit time stepping schemes is computationally expensive. Furthermore, as already pointed out, the choices of appropriate values of the regularization parameter α and the length of the time interval T are not easy. A computed example with a semi-implicit integration method is presented in Section 5.

The present paper describes new cascadic multilevel methods that share the computational efficiency of truncated iteration for the solution of linear discrete ill-posed problems (1.8) with the edge-preserving property of nonlinear models (1.5). The multilevel methods proceed from coarser to finer levels and are based on regularization by truncated iteration on each level. Prolongation of a coarse-level approximation of u to a finer grid is carried out by nonlinear edge-preserving and noise-reducing operators. Restrictions are computed by a local weighted least-squares method that preserves structures in the image. For many image restoration problems, the multilevel methods demand fewer matrix-vector product evaluations on the finest level than the corresponding 1-level truncated iterative methods and often determine restorations of higher quality. In fact, the quality of restorations obtained with multilevel methods can be higher than restorations computed by much more expensive methods for the solution of nonlinear models (1.5); see Section 5 as well as [20, Example 4.2].

This paper is organized as follows. Section 2 introduces the multilevel framework, reviews relevant properties of the iterative methods mentioned, and discusses a stopping criterion for the iterations on each level based on the discrepancy principle. This criterion requires that an estimate of the norm of the noise in the available contaminated image b^δ be available. Section 3 defines the restriction operators applied to the available noise- and blur-contaminated images, and shows a noise-reducing property, Section 4 describes the prolongation operators used and Section 5 presents numerical image restoration examples, which illustrate the performance of multilevel methods based on the LSQR, GMRES, and RRGMRES iterative methods. Section 5 also describes how an estimate of the norm of the noise in the data, b^δ, can be computed using the Perona-Malik nonlinear diffusion equation. It follows that the multilevel methods of the present paper also can be applied when no estimate for the norm of the noise in b^δ is explicitly known. Concluding remarks can be found in Section 6.

The application of multilevel methods to the solution of ill-posed problems is not new. Several solution methods for ill-posed problems apply a multilevel scheme to the regularized Tikhonov equations; see, e.g., Jacobsen et al. [17], Huckle and Staudacher [16], Zhu and Chen [32], and references therein. These methods reduce the computational effort, but do not improve the quality of the computed approximate solution, when compared with a standard iterative method applied on the finest level only. Multilevel methods that are applied directly to the equations (1.2) or (1.8), and regularize by truncated iteration, are described in [9, 10, 20, 25]. However, the methods [9, 10] are not regularization methods; see Section 2 for a discussion on the latter. The scheme in [20] is for symmetric problems and the approach in [25] is not tailored to image restoration. The present paper is the first one to discuss multilevel methods for the solution of nonsymmetric image restoration problems of the form (1.8).

2. Cascadic multilevel methods.

This section first describes the 1-level standard iterative methods used in the multilevel methods, discusses termination of the iterations by the discrepancy principle, and then presents the multilevel methods. The initial approximate solution, u_0, is assumed to be zero for all iterative methods. Introduce for $v = [v^{(1)}, v^{(2)}, \ldots, v^{(n)}]^T \in \mathbb{R}^n$ the weighted least-squares norm

$$\|v\| = \left(\frac{1}{n} \sum_{i=1}^{n} \left(v^{(i)} \right)^2 \right)^{1/2}. \tag{2.1}$$

GMRES is one of the most popular iterative methods for the solution of large-scale linear well-posed problems; see [28, Chapter 6] for a thorough treatment of this method. The jth iterate, u_j, determined by GMRES applied to the solution of (1.8) solves the minimization problem

$$\|Au_j - b^\delta\| = \min_{u\in\mathcal{K}_j(A,b^\delta)} \|Au - b^\delta\|, \qquad u_j \in \mathcal{K}_j(A, b^\delta),$$

where $\mathcal{K}_j(A, b^\delta) = \text{span}\{b^\delta, Ab^\delta, \dots, A^{j-1}b^\delta\}$ is a Krylov subspace. The computation of u_j requires the evaluation of j matrix-vector products with the matrix A.

The iterates u_j determined by the closely related RRGMRES method applied to the solution of (1.8) satisfy

$$\|Au_j - b^\delta\| = \min_{u\in\mathcal{K}_j(A,Ab^\delta)} \|Au - b^\delta\|, \qquad u_j \in \mathcal{K}_j(A, Ab^\delta).$$

Thus, RRGMRES and GMRES differ in the Krylov subspaces used; all iterates of RRGMRES live in the range of A. For many linear discrete ill-posed problems RRGMRES determines better approximations of the desired solution $\hat{u}$ of (1.9) than GMRES; see Section 5 for some computed examples, as well as [5] for a discussion. When A is symmetric, RRGMRES reduces to MR-II, an iterative method proposed and analyzed by Hanke [14, Chapter 6] for the solution of linear ill-posed problems with a symmetric indefinite operator A. The computation of u_j requires $j+1$ matrix-vector product evaluations with A.

LSQR is an implementation of the conjugate gradient method applied to the normal equations associated with (1.8); see [1, 23] for details. The jth iterate, u_j, determined by LSQR satisfies

$$\|Au_j - b^\delta\| = \min_{u\in\mathcal{K}_j(A^*A,A^*Ab^\delta)} \|Au - b^\delta\|, \qquad u_j \in \mathcal{K}_j(A^*A, A^*b^\delta),$$

where A^* denotes the adjoint of A. The computation of u_j requires the evaluation of j matrix-vector products with both A and A^*, giving a total of $2j$ matrix-vector product evaluations.

All the above iterative methods minimize the norm of residual error $b^\delta - Au_j$ over a Krylov subspace. Since the Krylov subspaces are nested, we have

$$\|Au_{j+1} - b^\delta\| \leq \|Au_j - b^\delta\|$$

for all the iterative methods considered. This property makes it natural to terminate the iterations by the discrepancy principle, defined below.

Let δ denote the norm of the error (1.10) in the right-hand side of (1.8), i.e.,

$$\delta = \|e^\delta\|, \tag{2.2}$$

and assume that a fairly accurate estimate of δ is known. This allows us to use the discrepancy principle to decide how many steps to carry out with the iterative methods. In Example 5.6 of Section 5, we discuss an approach to compute an estimate of δ for problems for which such an estimate is not explicitly available.

Definition (Discrepancy Principle). *Let $\gamma > 1$ be a fixed constant and let δ be given by (2.2). The vector u is said to satisfy the discrepancy principle if $\|b^\delta - Au\| \leq \gamma\delta$.*

We terminate the iterations as soon as an iterate u_j that satisfies the discrepancy principle has been computed.

Stopping Rule 2.1. *Let δ and γ be the same as in the Discrepancy Principle. Terminate the iterations when for the first time*

$$\|b^\delta - Au_j\| \leq \gamma\delta. \tag{2.3}$$

Denote the resulting stopping index by $j(\delta)$.

Note that, in general, $j(\delta)$ increases as δ decreases with γ kept fixed. An iterative method equipped with this stopping rule is said to be a regularization method if the computed iterates $u_{j(\delta)}$ satisfy

$$\lim_{\delta \searrow 0} \sup_{\|e^\delta\| \leq \delta} \|u_{j(\delta)} - \hat{u}\| = 0, \tag{2.4}$$

where $\hat{u}$ is the minimal-norm solution of (1.9). Hanke [14, Theorem 6.15] and Nemirovskii [21] have shown that LSQR is a regularization method in a Hilbert space setting. Calvetti et al. [6] show that GMRES is a regularization method under more stringent conditions. The latter proof carries over to RRGMRES. In particular, all the iterative methods mentioned satisfy (2.3) in finite dimension in the absence of breakdown of GMRES and RRGMRES. How to circumvent breakdown is discussed in [26]; see also below for comments on the latter.

We are in a position to present multilevel methods based on the iterative methods described above. A termination criterion similar to Stopping Rule 2.1 is applied on each level. Let

$$W_1 \subset W_2 \subset \cdots \subset W_\ell$$

be a sequence of nested subspaces of $\mathbb{R}^n$ of dimensions $\dim(W_i) = n_i$ with $n_1 < n_2 < \ldots < n_\ell = n$. We refer to the subspaces W_i as levels, with W_1 being the coarsest and $W_\ell = \mathbb{R}^n$ the finest level. Each level is furnished with a weighted least-squares norm; level W_i has a norm of the form (2.1) with n replaced by n_i.

This section discusses multilevel methods for image restoration for which we let

$$n_{i-1} = \frac{1}{4} n_i, \qquad 1 < i \leq \ell. \tag{2.5}$$

For notational simplicity, we assume the image to be square, i.e., the image is represented on level i by a square array of pixels $p_i^{(j,k)}$, $1 \leq j, k \leq \sqrt{n_i}$; here j and k denote the horizontal and vertical coordinates, respectively.

Let $A_i \in \mathbb{R}^{n_i \times n_i}$ be the representation of the blurring operator A on level W_i. The matrix A_i is determined by discretization of the integral operator (1.2) similarly as A. This defines implicitly the restriction operator $R_i : \mathbb{R}^n \to W_i$, such that

$$A_i = R_i A R_i^*. \tag{2.6}$$

Some discretizations require that the adjoint R_i^* be replaced by another operator $Q_i : W_i \to \mathbb{R}^n$; however, this is not the case for the computed examples of Section 5. We define $R_\ell = I$.

The choice of restriction operators R_i for determining the restrictions A_i of A is in our experience less crucial for achieving high-quality restorations than the choice of restriction operators $R_i^{(\omega)} : \mathbb{R}^n \to W_i$ for reducing the available blur- and noise-contaminated image represented by the right-hand side b^δ in (1.8); thus, we let

$$b_i^\delta = R_i^{(\omega)} b^\delta, \qquad 1 \leq i < \ell, \tag{2.7}$$

where the $R_i^{(\omega)}$ are determined by repeated local weighted least-squares approximation, inspired by a "staircasing"-reducing scheme recently proposed by Buades et al.

[3]. The precise definition of the $R_i^{(\omega)}$ and a discussion of their noise-reducing property are provided in Section 3. We let $R_\ell^{(\omega)} = I$.

The choice of prolongation operators from level $i-1$ to level i is important for the performance of the multilevel method. We apply nonlinear prolongation operators $P_i : W_{i-1} \to W_i$, $1 < i \leq \ell$, defined by piecewise linear interpolation followed by integration of the Perona-Malik equation over a short time interval; see Section 4 for details on the latter. These operators are designed to be noise-reducing and edge-preserving. For comparison, we also consider linear prolongation operators $L_i : W_{i-1} \to W_i$, $1 < i \leq \ell$, defined by piecewise linear interpolation, and linear prolongation operators $\widehat{L}_i : W_{i-1} \to W_i$, $1 < i \leq \ell$, obtained by linearization of the Perona-Malik diffusion equation. Computed examples reported in Section 5 show the nonlinear prolongation operators P_i to yield more accurate restorations than the linear operators L_i and $\widehat{L}_i$.

The multilevel methods of the present paper are cascadic, i.e., they first determine an approximate solution of $A_1 u = b_1^\delta$ in W_1 using one of the iterative methods GMRES, RRGMRES, or LSQR. We refer to the iterative method as IM in Algorithm 2.2 below. The iterations with this method are terminated as soon as an iterate that satisfies a stopping rule related to the discrepancy principle has been determined. This iterate is mapped from W_1 into W_2 by the prolongation P_2. A correction of this mapped iterate in W_2 is computed by IM. Again, the iterations are terminated by a stopping rule related to the discrepancy principle. The approximate solution in W_2 determined in this manner is mapped into W_3 by P_3. The computations are continued in this fashion until an approximation of $\hat{u}$ has been determined in $W_\ell = \mathbb{R}^n$. The latter approximation is post-processed by local weighted least-squares approximation, as suggested by Buades et al. [3], in order to reduce staircasing. This smoothing step is denoted by the operator $S_\ell : W_\ell \to W_\ell$ in Algorithm 2.2, which outlines the computations for the multilevel methods.

ALGORITHM 2.2. *Multilevel Algorithm*
Input: A, b^δ, δ, $\ell \geq 1$ *(number of levels);*
Output: approximate solution $u_\ell \in W_\ell$ *of (1.8);*
Determine A_i *and* b_i^δ *from (2.6) and (2.7), respectively, for* $1 \leq i \leq \ell$*;*
$u_0 := 0$*;*
for $i := 1, 2, \ldots, \ell$ *do*
  $u_{i,0} := P_i u_{i-1}$*;*
  $\Delta u_{i,m_i} := IM(A_i, b_i^\delta - A_i u_{i,0})$*;*
  Correction step: $u_i := u_{i,0} + \Delta u_{i,m_i}$*;*
endfor
$u_\ell := S_\ell u_\ell$*;* □

In the algorithm $\Delta u_{i,m_i} := IM(A_i, b_i^\delta - A_i u_{i,0})$ denotes the computation of the approximate solution $\Delta u_{i,m_i}$ of

$$A_i z_i = b_i^\delta - A_i u_{i,0} \tag{2.8}$$

by m_i iterations with one of the iterative methods GMRES, RRGMRES, or LSQR, using the initial iterate $\Delta u_{i,0} = 0$. We remark that the structure of Algorithm 2.2 is the same as that of [20, Algorithm 2.2]; however, the iterative methods, as well as the prolongation, restriction, and smoothing operators applied in the algorithms differ. Cascadic multilevel methods related to those of Algorithm 2.2 also are discussed in [25]. Again, the methods differ in the choices of prolongation and restriction operators.

Moreover, no smoothing operator S_ℓ is applied in [25].

Let

$$\hat{b}_i = R_i^{(\omega)}\hat{b}, \qquad 1 \le i \le \ell, \tag{2.9}$$

i.e., $\hat{b}_i$ is the (unavailable) representation of the blurred but noise-free image $\hat{b}$ on level i, with $\hat{b}_\ell = \hat{b}$. Our stopping rule on each level is based on the assumption that there are constants c_i independent of δ, such that

$$\|b_i^\delta - \hat{b}_i\| \le c_i\delta, \qquad 1 \le i \le \ell, \tag{2.10}$$

where δ satisfies (2.2). The noise-reducing property of the restriction operators $R_i^{(\omega)}$, see (3.9) in Section 3, suggests the choice

$$c_i = \frac{1}{3}c_{i+1}, \qquad 1 \le i < \ell, \qquad c_\ell = \gamma, \tag{2.11}$$

where γ is the same as in (2.3).

Stopping Rule 2.3. *Let δ and the c_i be the same as in (2.10) and denote the iterates determined by the IM iterative method applied to the solution of (2.8) with initial iterate $\Delta u_{i,0} = 0$ by $\Delta u_{i,m}$, $m = 1, 2, \ldots$. Terminate the iteration on level i as soon as an iterate $\Delta u_{i,m_i}$ that satisfies*

$$\|b_i - A_i u_{i,0} - A_i \Delta u_{i,m_i}\| \le c_i\delta$$

has been determined, where $m_i = m_i(\delta)$ denotes the termination index.

Since images are represented by pixels, image restoration problems live in finite dimensional spaces. It is quite straightforward to show that the LSQR-based multilevel method defined by Algorithm 2.2, with linear prolongation operators P_i and $S_\ell = I$, is a regularization method for finite dimensional problems. The P_i may, for instance, be the operators L_i defined by piecewise linear interpolation. Alternatively, we may define the P_i by time integration of a linearized Perona-Malik diffusion equation; see Section 4 for more details on the latter. In Theorem 2.4 and below, $\mathcal{R}(M)$ and $\mathcal{N}(M)$ denote the range and null space, respectively, of the matrix M.

Theorem 2.4. *Let, for $1 \le i \le \ell$, the equation*

$$A_i u = \hat{b}_i \tag{2.12}$$

be consistent with minimal-norm solution $\hat{u}_i$. In particular, $\hat{u}_\ell = \hat{u}$. Let the prolongation operator P_i in Algorithm 2.2 be linear for all i and let $S_\ell = I$ in the last step of Algorithm 2.2. Assume that

$$\mathcal{R}(P_i) \subset \mathcal{R}(A_i^*), \qquad 1 < i \le \ell. \tag{2.13}$$

Let the LSQR-based Algorithm 2.2 determine approximate solutions of the normal equations associated with the linear systems (2.8) for $1 \le i \le \ell$. Assume that the right-hand sides of the normal equations are generic, i.e., they do not belong to an invariant subspace of low dimension. Let the errors in the right-hand sides restricted to W_i be of the form

$$b_i^\delta - \hat{b}_i = c_i\delta\, w_i, \qquad 1 \le i \le \ell, \tag{2.14}$$

where $w_i \in W_i$ is a unit vector independent of δ, and the c_i are the positive constants in (2.10). Terminate the LSQR-iterations in Algorithm 2.2 on each level according to

Stopping Rule 2.3 to obtain the iterates $u_i \in W_i$ for $1 \leq i \leq \ell$. The multilevel method so defined is a regularization method on each level, i.e.,

$$\lim_{\delta \searrow 0} \|u_i - \hat{u}_i\| = 0, \qquad 1 \leq i \leq \ell. \tag{2.15}$$

Proof. The proof of Theorem 2.4 in [20] is sufficiently general to allow adaption to the present situation. The constants $c_i > 0$ can be chosen arbitrarily. If we only require (2.15) to hold for $i = \ell$, then the requirements of the theorem can be relaxed. For instance, then we only have to require that (2.13) holds for $i = \ell$. □

REMARK 2.1. *The fact that an iterative method is a regularization method in the sense that it satisfies (2.15) does not guarantee that it produces accurate restorations. However, methods that violate (2.15) may yield poor restorations when the noise e^δ in b^δ is of small norm δ; cf. (2.2).*

Results related to Theorem 2.4 in infinite dimensions are discussed in [25, Theorem 3.2 and Corollary 3.1]. Donatelli and Serra [9, 10] describe noncascadic multilevel methods, in which full V- or W-cycles are carried out. These methods are in [10] shown not to be regularization methods in the sense of the present paper; nevertheless, the computed examples presented in [9, 10] look nice.

When (2.13) does not hold for the index i, the computed iterate u_i may have a component in $\mathcal{N}(A_i)$ and, therefore, not converge to the minimal-norm solution of (2.12) as $\delta \searrow 0$. Numerical experience indicates that the condition (2.13) does not pose a practical problem; in all our experiments, Algorithm 2.2 computed iterates u_i that are numerically orthogonal to $\mathcal{N}(A_i)$ for all i.

A matrix M is said to be range symmetric if $\mathcal{R}(M) = \mathcal{R}(M^*)$. Theorem 2.4 also holds for RRGMRES-based multilevel methods when the matrices A_i are range symmetric or nonsingular. When A_i is singular and not range symmetric, the multilevel method may determine an approximation of $\hat{u}_i$ that has a component in $\mathcal{N}(A_i)$. GMRES-based multilevel methods are only guaranteed to determine approximations of $\hat{u}_i$ orthogonal to $\mathcal{N}(A_i)$ when A_i is nonsingular. Implementations of GMRES and RRGMRES for linear systems of equations with a singular matrix are discussed in [26].

3. Noise-reducing restriction operators. The nonlinear model (1.5) based on the Perona-Malik regularization operator defined by (1.6) with g given by (1.7) may determine restored images that suffer from so-called staircasing, i.e., the computed images display large flat regions separated by artificial boundaries, instead of a desired smooth surface. Recently, Buades et al. [3] proposed post-processing of restored images by local weighted least-squares approximation in order to reduce this effect. We define the mappings $M_i^{(\omega)} : W_i \to W_{i-1}$ by solving these least-squares problems as follows. Let p_{i-1} denote the image on level $i-1$ of size $\sqrt{n_{i-1}} \times \sqrt{n_{i-1}}$ obtained by restriction of the image p_i on level i of size $\sqrt{n_i} \times \sqrt{n_i}$. Superscripts denote pixel coordinates, i.e., $p_i^{(j,k)}$ denotes the pixel with coordinates (j,k) of p_i. For each pixel $p_{i-1}^{(j,k)}$ of the image p_{i-1}, we solve a weighted least-squares problem using a 3×3 window of neighboring pixels. This window and the coordinates of the pixels involved are displayed by Figure 3.1. Pixels $p_{i-1}^{(j,k)}$ of the coarser image p_{i-1} are shown with gray background.

Introduce the weight function

$$\omega_i^{(2j,2k)}(s,t) = \exp\left(-\kappa \left(p_i^{(2j+s,2k+t)} - p_i^{(2j,2k)}\right)^2\right), \tag{3.1}$$

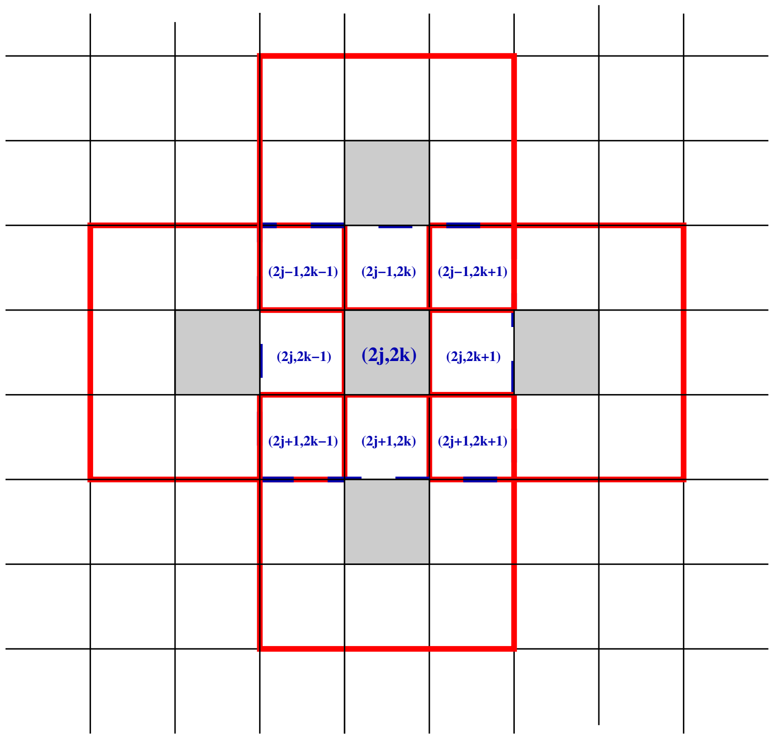


FIG. 3.1. *Pixel mask used in the local weighted least-squares computation.*

with κ a positive constant, and consider the local weighted least-squares problem

$$\min_{\alpha_0,\alpha_1,\alpha_2} \sum_{s,t\in\{0,\pm1\}} \left(p_i^{(2j+s,2k+t)} - (\alpha_0 + \alpha_1 s + \alpha_2 t)\right)^2 \omega_i^{(2j,2k)}(s,t) \tag{3.2}$$

for the coefficients $\{\alpha_0, \alpha_1, \alpha_2\}$ of the linear function. Let $\{\hat\alpha_0, \hat\alpha_1, \hat\alpha_2\}$ denote the solution and define $p_{i-1}^{(j,k)} := \hat\alpha_0$. We solve the least-squares problems (3.2) for all $1 \le j, k \le \sqrt{n_{i-1}}$. This defines $M_i^{(\omega)}$.

The number of arithmetic floating point operations required to solve (3.2) for all pixels in W_{i-1} is proportional to n_{i-1} and therefore modest. The solution of each least-squares problem can be computed, e.g., by determining a modified QR-decomposition of a 9×3 matrix based on modified Householder transformations; see Gulliksson and Wedin [12] for details on the latter. The modifications are required because of the weights in the least-squares problem.

Application of the mappings $M_k^{(\omega)}$ for $k = \ell, \ell+1, \ldots, i+1$, in order, defines the weighted restriction operators $R_i^{(\omega)} : \mathbb{R}^n \to W_i$ according to

$$R_i^{(\omega)} = M_{i+1}^{(\omega)} M_{i+2}^{(\omega)} \ldots M_\ell^{(\omega)}, \qquad 1 \le i < \ell. \tag{3.3}$$

Applying the mapping $R_i^{(\omega)}$ to the right-hand side of (1.8) yields the vectors b_i^δ; cf. (2.7). The restriction operators (3.3) typically yield more faithful restorations than simpler restriction operators defined by local averaging of neighboring pixel-values; see Example 5.2 of Section 5 for an illustration.

Since least-squares approximation implies smoothing and the noise $e_\ell^\delta = e^\delta$ in $b_\ell^\delta = b^\delta$ contains a significant high-frequency component, the restrictions b_i^δ of b^δ, for $i < \ell$, defined by (2.7) typically contain less noise than b_ℓ^δ. This is illustrated by the following result.

Proposition 3.1. *Let $\kappa = 0$ in (3.1) and assume that the entries $e_\ell^{(j,k)}$, $1 \leq j,k \leq \sqrt{n_\ell}$, of the noise-vector e_ℓ^δ are uncorrelated random variables with zero mean. Let $\|e_\ell^\delta\|_V^2$ denote the average variance of the entries, i.e.,*

$$\|e_\ell^\delta\|_V^2 = \frac{1}{n_\ell} \sum_{j,k=1}^{\sqrt{n_\ell}} Var(e_\ell^{(j,k)}).$$

Assume that the image is periodic, so that boundary effects can be ignored, and let

$$e_{\ell-1}^\delta = R_{\ell-1}^{(\omega)} e_\ell^\delta.$$

Then

$$\|e_{\ell-1}^\delta\|_V \leq \frac{4}{9}\|e_\ell^\delta\|_V, \qquad \|e_{\ell-1}^\delta\|_V \geq \frac{2}{9}\|e_\ell^\delta\|_V.$$

Proof. For $i = \ell$, the least-squares problem (3.2) with $\omega^{(2j,2k)} = 1$ for all j can be expressed as

$$\min_{\alpha \in \mathbb{R}^3} \|H\alpha - q^{(j,k)}\| \tag{3.4}$$

with

$$H = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & -1 & -1 & -1 \\ 0 & 1 & -1 & 0 & 1 & -1 & 0 & 1 & -1 \end{bmatrix}^T \in \mathbb{R}^{9\times 3},$$

$$\alpha = [\alpha_0, \alpha_1, \alpha_2]^T \in \mathbb{R}^3,$$

$$q^{(j,k)} = [p_\ell^{(2j,2k)}, p_\ell^{(2j+1,2k)}, p_\ell^{(2j-1,2k)}, \ldots, p_\ell^{(2j-1,2k-1)}]^T \in \mathbb{R}^9.$$

Denote the solution of (3.4) by $\hat{\alpha} = [\hat{\alpha}_0, \hat{\alpha}_1, \hat{\alpha}_2]^T$. Since H has orthogonal columns, we obtain

$$p_{\ell-1}^{(j,k)} = \hat{\alpha}_0 = \frac{1}{9} \sum_{s,t \in \{0,\pm 1\}} p_\ell^{(2j+s,2k+t)}.$$

The component $e_{\ell-1}^{(j,k)}$ of the vector $e_{\ell-1}^\delta$ represents the noise in $p_{\ell-1}^{(j,k)}$ and is the average of the errors $e_\ell^{(2j+s,2k+t)}$ in the nine pixels $p_\ell^{(2j+s,2k+t)}$, $s,t \in \{0, \pm 1\}$. Therefore,

$$\mathrm{Var}(e_{\ell-1}^{(j,k)}) = \frac{1}{81} \sum_{s,t \in \{0,\pm 1\}} \mathrm{Var}(e_\ell^{(2j+s,2k+t)})$$

and it follows that

$$\|e_{\ell-1}^\delta\|_V^2 = \frac{1}{n_{\ell-1}} \sum_{j,k=1}^{\sqrt{n_{\ell-1}}} \mathrm{Var}(e_{\ell-1}^{(j,k)}) \tag{3.5}$$

$$= \frac{1}{81 n_{\ell-1}} \sum_{j,k=1}^{\sqrt{n_{\ell-1}}} \sum_{s,t\in\{0,\pm1\}} \mathrm{Var}(e_\ell^{(2j+s,2k+t)}) \tag{3.6}$$

$$\leq \frac{4}{81 n_{\ell-1}} \sum_{j,k=1}^{\sqrt{n_\ell}} \mathrm{Var}(e_\ell^{(j,k)}) = \frac{16}{81 n_\ell} \sum_{j,k=1}^{\sqrt{n_\ell}} \mathrm{Var}(e_\ell^{(j,k)}) = \frac{16}{81} \|e_\ell^\delta\|_V^2. \tag{3.7}$$

The inequality (3.7) is a consequence of the fact that some errors $e_\ell^{(2j+s,2k+t)}$ contribute to four errors $e_{\ell-1}^{(j,k)}$. This can be seen from Figure 3.1. The center pixel in the figure is in $W_{\ell-1}$ and gets contributions from the four corner pixels on level W_ℓ. Moreover, no pixel in W_ℓ contributes to more than four pixels in $W_{\ell-1}$. The equality on the left in (3.7) follows from (2.5).

Every pixel in W_ℓ contributes to at least one pixel in $W_{\ell-1}$. We therefore obtain analogously to (3.5)-(3.7) the inequality

$$\begin{aligned} \|e_{\ell-1}^\delta\|_V^2 &= \frac{1}{81 n_{\ell-1}} \sum_{j,k=1}^{\sqrt{n_{\ell-1}}} \sum_{s,t\in\{0,\pm1\}} \mathrm{Var}(e_\ell^{(2j+s,2k+t)}) \\ &\geq \frac{1}{81 n_{\ell-1}} \sum_{j,k=1}^{\sqrt{n_\ell}} \mathrm{Var}(e_\ell^{(j,k)}) = \frac{4}{81 n_\ell} \sum_{j,k=1}^{\sqrt{n_\ell}} \mathrm{Var}(e_\ell^{(j,k)}) = \frac{4}{81} \|e_\ell^\delta\|_V^2. \end{aligned}$$

This establishes the proposition. □

Corollary 3.2. *Let the conditions of Proposition 3.1 hold and assume further that*

$$Var(e_\ell^{(j,k)}) = \eta^2, \qquad 1 \leq j,k \leq \sqrt{n_\ell}. \tag{3.8}$$

Then

$$\|e_{\ell-1}^\delta\|_V = \frac{1}{3} \|e_\ell^\delta\|_V.$$

Proof. The result follows from (3.5)-(3.6). □

Approximating $\|e_k^\delta\|_V$ by $\|e_k^\delta\|$ for $k = \ell-1, \ell$, we obtain from Corollary 3.2 that

$$\|e_{\ell-1}^\delta\| \approx \frac{1}{3} \|e_\ell^\delta\|. \tag{3.9}$$

We will use the factor $1/3$ in our multilevel method; cf. (2.11).

4. Edge-preserving nonlinear prolongation operators. We describe the nonlinear edge-preserving prolongation operators P_i used in the computed examples. They have previously been applied in [20], where further details on their implementation are provided; see also [30].

The prolongation operators consist of two parts: first the approximate solution on level W_{i-1} determined in the correction step of Algorithm 2.2 is mapped into W_i by piecewise linear interpolation. This is carried out by the operator L_i. The element $L_i u_{i-1}$ in W_i then is used as initial function for a discretization of the initial-boundary value problem for the Perona-Malik nonlinear diffusion equation

$$\frac{\partial u}{\partial t} = \mathrm{div}(g(|\nabla u|^2)\nabla u), \tag{4.1}$$

where g is the Perona-Malik diffusivity (1.7). Integration over a short time interval removes noise while preserving rapid spacial transitions, such as edges. Discretization of (4.1) in space yields the initial value problem

$$\frac{du}{dt} = L(u)u, \qquad u \in W_i \qquad t > 0, \tag{4.2}$$

with initial function $u^0 = L_i u_{i-1}$. We set u to zero on the boundary. Here $L(u)u$ is the same discretization of the Perona-Malik operator as in (1.11). The differential equation (4.2) is designed to determine an element in W_i that has edges close to those of u_{i-1} in W_{i-1}.

The matrix $L(u)$ in (4.2) has, generically, five nonvanishing entries in each row. This makes the evaluation of $L(u)u$ quite inexpensive. Integration is performed by carrying out about 10 time steps of size about 0.2 with an explicit method. The small number of time steps avoids difficulties due to numerical instability and keeps the computational work required for integration negligible. We found it to be beneficial to apply more time steps the more noise-contaminated the available image. However, in our experience the exact choices of the number of time steps and of the size of the time steps is not crucial for the good performance of the multilevel methods.

The proof of Theorem 2.4 requires the prolongation operators to be linear. We therefore also consider the discrete linearized Perona-Malik diffusion equation

$$\frac{du}{dt} = L(u^0)u, \tag{4.3}$$

where $u^0 = L_i u_{i-1}$. Integration over a short time interval with initial value u^0 defines the linear prolongation operator $\widehat{L}_i$. Example 5.2 of Section 5 shows the operators $\widehat{L}_i$ to yield restored images of almost the same quality as the prolongation operators P_i. However, we do not advocate the use of the linearized operators $\widehat{L}_i$ instead of the nonlinear operators P_i since integration of (4.2) is fast and easy.

This concludes the description of the components of Algorithm 2.2. Thus, the restriction operators are defined as described in Section 3 and are based on a local weighted least-squares approximation scheme described by Buades et al. [3]. The latter also is used for the smoother S_ℓ on the finest level. Prolongation is carried out as described in the present section. Our multilevel methods seek to reduce the blur by solving linear systems of equations on each level, and to reduce the noise by integrating a discretization of the nonlinear diffusion equation (4.1). We remark that noise-reduction also can be achieved by other means, such as by integrating a differential equation based on the total variation-norm. This is discussed in [20]. A comparison of the latter approach with the one of the present paper reported in [20] showed integration of (4.1) to yield restored images of slightly higher quality.

5. Computed examples. We illustrate the performance of Algorithm 2.2 applied to the restoration of two-dimensional gray-scale images that have been contaminated by blur and noise. The images are represented by arrays of 8-bit pixels. The computed examples compare cascadic multilevel methods based on the GMRES, RRGMRES, and LSQR iterative methods.

Define the noise-level

$$\nu = \frac{\|e^\delta\|}{\|\hat{b}\|}. \tag{5.1}$$

We assume that an accurate estimate of ν is available in all examples of this section, except for Example 5.6, and therefore choose the parameter γ in (2.3) and (2.11) close to unity; specifically, we let $\gamma = 1.01$. Examples 5.1 and 5.2 discuss restoration of images that have been contaminated by space-variant Gaussian blur and noise, while Examples 5.3 and 5.4 are concerned with images that are corrupted by motion blur and noise. In Example 5.5 we compare the multilevel method to the nonlinear model (1.5). We illustrates how the noise-level can be estimated by integrating the Perona-Malik diffusion equation over a short time interval in Example 5.6. Estimates so obtained are used in the stopping rule of Algorithm 2.2 and we compare the performance of the algorithm for exact and estimated noise-levels.

The matrices A_i defined by (2.6) do not have to be explicitly stored; it suffices to define functions for the evaluation of matrix-vector products with the A_i. For the examples of the present section, these products can be computed efficiently by using the structure of the A_i; see, e.g., [22, 15] for discussions. The matrices corresponding to the finest level are numerically singular in all examples.

The displayed restored images provide a qualitative comparison of the performance of the proposed cascadic multilevel methods. The Peak Signal-to-Noise Ratio (PSNR),

$$\text{PSNR}(u_\ell, \hat{u}) = 20 \log_{10} \frac{255}{\|u_\ell - \hat{u}\|} \text{ dB}, \tag{5.2}$$

where $\hat{u}$ is the blur- and noise-free image and u_ℓ the restored image determined by Algorithm 2.2, provides a quantitative comparison. The norm $\|u_\ell - \hat{u}\|$ is the Root Mean Squared Error (RMSE) of $u_\ell - \hat{u}$; cf. (2.1). The numerator 255 is the largest pixel-value that can be represented with 8 bits. A high PSNR-value indicates that the restoration is accurate; however, the PSNR-values are not always in agreement with visual perception. The computations are carried out in MATLAB with about 16 significant decimal digits.

		GMRES		RRGMRES		LSQR	
ℓ	ν	PSNR	# iter	PSNR	# iter	PSNR	# iter
1	$5 \cdot 10^{-3}$	33.78	10	31.45	4	33.92	13
2	$5 \cdot 10^{-3}$	36.35	8 9	33.25	4 3	35.84	8 10
3	$5 \cdot 10^{-3}$	36.41	3 8 9	32.78	3 3 3	35.86	3 8 10
1	$1 \cdot 10^{-2}$	31.89	6	29.77	3	32.36	9
2	$1 \cdot 10^{-2}$	34.29	5 5	31.14	3 2	34.09	5 7
3	$1 \cdot 10^{-2}$	33.97	3 4 5	31.48	3 3 2	33.96	3 4 7
1	$5 \cdot 10^{-2}$	26.00	2	27.98	2	28.74	3
2	$5 \cdot 10^{-2}$	28.96	2 1	28.80	2 1	29.90	2 2
3	$5 \cdot 10^{-2}$	27.10	3 1 2	29.16	3 2 1	30.23	3 2 2
1	$1 \cdot 10^{-1}$	22.09	2	24.79	1	27.48	2
2	$1 \cdot 10^{-1}$	27.50	2 1	28.69	2 1	28.50	2 1
3	$1 \cdot 10^{-1}$	27.60	3 1 1	27.99	3 1 1	29.21	3 1 2

TABLE 5.1

Example 5.1. Restoration of corrupted versions of the `corner` *image. The table reports PSNR-values of the restored images determined and the number of iterations (# iter) required for several noise-levels ν and several cascadic multilevel methods. The 1-level methods are standard iterative schemes applied on the finest level only. The point spread function is nonsymmetric and Gaussian with* `band` $= 7$ *and* $\sigma_1 = 4$, $\sigma_2 = 1$.

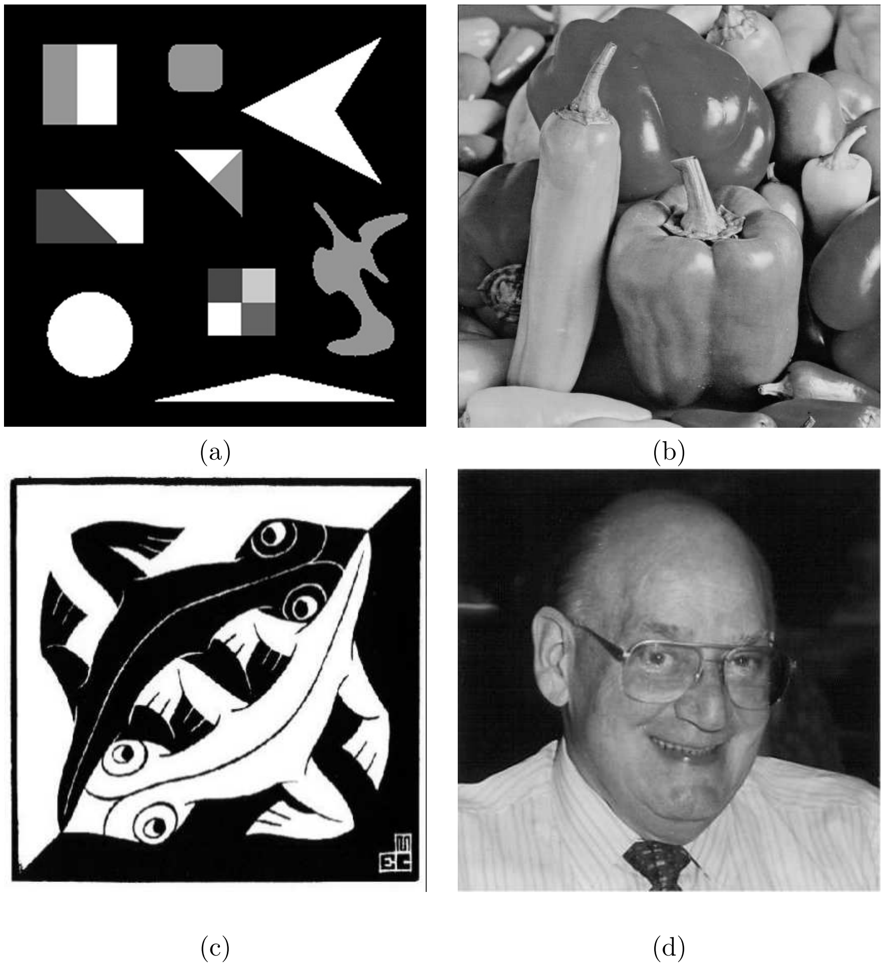


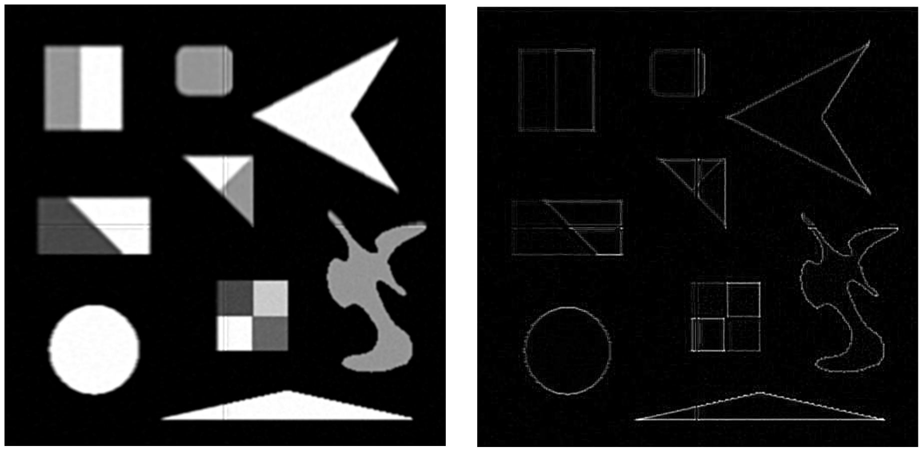
FIG. 5.1. *Blur- and noise-free images used in the numerical experiments: (a)* `corner`*,* 512×512 *pixels; (b)* `pepper`*,* 512×512 *pixels; (c)* `lizard`*,* 412×412 *pixels; (d)* `Varga`*,* 360×360 *pixels.*

Example 5.1. We compare image restorations determined by Algorithm 2.2 with the GMRES, RRGMRES, and LSQR iterative methods and nonlinear prolongation. The number of iterations is determined by Stopping Rules 2.1 or 2.3. We determine restorations of contaminated versions of the original 512×512-pixel `corner` image displayed in Figure 5.1(a). The contamination is caused by noise and spatially variant Gaussian blur. The nonsymmetric blurring matrix is given by

$$A := I_1(T_1 \otimes T_1) + I_2(T_2 \otimes T_2) \in \mathbb{R}^{n \times n}, \qquad n = 512^2.$$

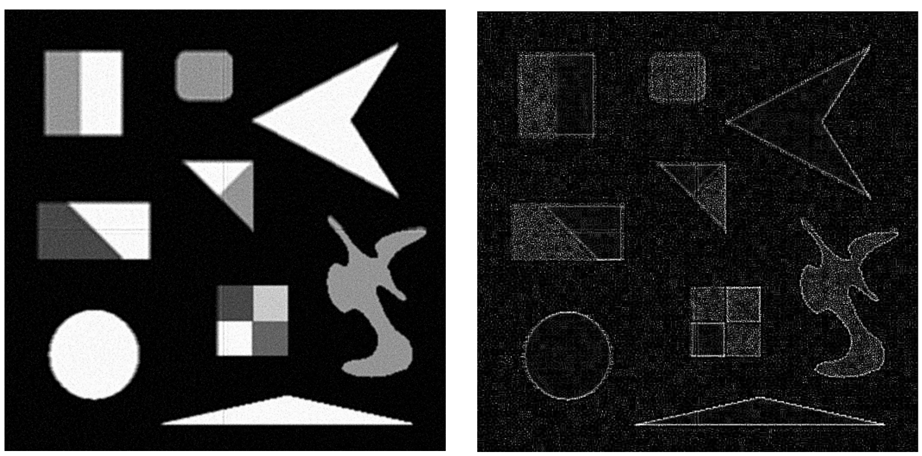
Here I_1 is the diagonal matrix, whose first $n/2$ diagonal entries are one, and the remaining entries zero, and $I_2 := I - I_1$. The operator $\otimes$ denotes the Kronecker product and the T_ℓ are banded Toeplitz matrices, which represent Gaussian blur in one space-dimension. Specifically, $T_\ell = [t_{jk}^{(\ell)}]_{j,k=1}^{\sqrt{n}}$ is defined by

$$t_{jk}^{(\ell)} := \begin{cases} \frac{1}{\sigma_\ell \sqrt{2\pi}} \exp\left(-\frac{(j-k)^2}{2\sigma_\ell^2}\right), & \text{if } |j-k| \leq \texttt{band}, \\ 0, & \text{otherwise.} \end{cases}$$



(a) (b)

FIG. 5.2. *Example 5.1. (a) Restoration determined by RRGMRES of blur- and noise-contaminated* `corner` *image with noise-level* $\nu = 5 \cdot 10^{-2}$*, and (b) edge-map for restoration.*



(a) (b)

FIG. 5.3. *Example 5.1. (a) Restoration determined by GMRES of blur- and noise-contaminated* `corner` *image with noise-level* $\nu = 5 \cdot 10^{-2}$*, and (b) edge-map for restoration.*

This models the situation when the left and right halves of the image are degraded by different Gaussian blurs. The matrix A is nonsymmetric; its first $n/2$ rows are the made up of the first $n/2$ rows of $T_1 \otimes T_1$ and its last $n/2$ rows are the last $n/2$ rows of $T_2 \otimes T_2$.

The parameter `band` determines the half bandwidth of the Toeplitz matrices applied. Enlarging `band` increases the storage requirement, the arithmetic work required for the evaluation of matrix-vector products with A, and to some extent the blurring. In this examples, we let `band` $= 7$ and $\sigma_1 = 4$, $\sigma_2 = 1$.

We first illustrate properties of the GMRES, RRGMRES, and LSQR iterative methods when used as 1-level methods. Throughout this section, 1-level method refers to a basic iterative method applied on the finest level only without smoothing. We apply the methods to restore the contaminated image. The restoration determined by RRGMRES is shown in Figure 5.2(a); Figure 5.2(b) displays the edge map for the restored image. This map is determined by the edge-detector of `gimp`, a public domain software tool for image processing.[1] The restored image determined by LSQR and the associated edge map are visually indistinguishable from those obtained with RRGMRES; they therefore are not shown. Figure 5.3(a) shows the restoration determined by GMRES and Figure 5.3(b) depicts the edge-map for the restoration. While the restorations determined by RRGMRES and GMRES look fairly similar when displayed in the size shown, the associated edge-maps differ; the edge-map for the restoration computed by GMRES shows less sharpness and much more noise.

PSNR-values for the restored images and the number of iterations required by 1-level methods for several noise-levels are reported in the first row of each horizontal block of Table 5.1. The table shows 1-level LSQR to determine the most accurate restorations for any noise-level. This 1-level method is the computationally most expensive one, because it requires the largest number of iterations and each iteration demands the evaluation of two matrix-vector products; GMRES and RRGMRES only require the evaluation of one matrix-vector product per iteration.

Table 5.1 shows ℓ-level GMRES, for $\ell > 1$, to give accurate restorations for noise-levels $\nu < 5 \cdot 10^{-2}$ with less computational effort than ℓ-level LSQR. However, ℓ-level RRGMRES and LSQR produce better restorations for high noise-levels $\nu \geq 5 \cdot 10^{-2}$.

The ℓ-level GMRES method often requires fewer iterations than ℓ-level RRGMRES, but generally delivers restorations of worse quality. Images restored by GMRES typically maintain sharp edges, but suffer from propagated noise and some "ringing" near the borders. This is illustrated by the edge map of Figure 5.3(b) for 1-level GMRES. For small noise-levels, the PSNR-values for images restored by ℓ-level GMRES are high; however, restorations determined by ℓ-level RRGMRES generally are more visually pleasing, even when they have lower PSNR-values.

Table 5.1 shows restorations obtained by cascadic multilevel methods to be of higher quality, as measured by the PSNR-values, than restorations computed by 1-level methods. This is in agreement with visual perception.

The columns labeled "# iter" display the number of iterations on each level. The ℓ-tupplets show, from left to right, the number of iterations for increasing level index. Thus, the leftmost entry shows the number of iterations on the coarsest level and the rightmost entry the number of iterations on the finest level. The number of required iterations can be seen to increase as the noise level decreases, both for 1-level and multilevel methods.

Because of the noted poorer quality of restorations determined by ℓ-level GMRES, $\ell \geq 1$, when compared with restored images determined by ℓ-level RRGMRES and ℓ-level LSQR, we omit ℓ-level GMRES methods in the remainder of this section. □

Example 5.2. We compare Algorithm 2.2 for piecewise linear and nonlinear prolongation operators. Tables 5.2-5.6 report the performance of the algorithm when used with the iterative methods RRGMRES or LSQR. The algorithm is applied to blur- and noise-contaminated versions of the images shown in Figure 5.1(a) and (c). The columns with header "L_i" show results obtained with the piecewise linear prolongation operators L_i without smoothing in the last step of the algorithm. Columns labeled "P_i" display results obtained when the nonlinear prolongation operators P_i are used and smoothing is applied in the last step of the algorithm. Similarly, the columns labeled "$\widehat{L}_i$" in Tables 5.2 and 5.3 show the performance of Algorithm 2.2 with the linearized prolongation operators $\widehat{L}_i$ described in Section 4 and no smooth-

[1]`gimp` is described and available at `http://www.gimp.org/`

		L_i		P_i		$\widehat{L}_i$	
ℓ	`band`	PSNR	# iter	PSNR	# iter	PSNR	# iter
1	5	29.75	2	-	-	-	-
2	5	29.40	2 1	30.06	2 1	29.93	2 1
3	5	29.92	3 2 1	30.55	3 2 1	30.41	3 2 1
1	7	28.02	2	-	-	-	-
2	7	28.25	2 1	28.80	2 1	28.67	2 1
3	7	27.88	3 1 1	29.16	3 2 1	28.99	3 2 1
1	9	26.80	2	-	-	-	-
2	9	26.87	2 1	27.30	2 1	27.24	2 1
3	9	27.25	7 1 1	28.07	7 1 1	27.98	7 1 1

TABLE 5.2

Example 5.2. Restoration of corrupted versions of the `corner` image by RRGMRES-based cascadic multilevel methods. The table reports PSNR-values of the restored images and the number of iterations (# iter) required for piecewise linear prolongation operators (L_i), nonlinear prolongation operators (P_i), and linearized prolongation operators described in Section 4 ($\widehat{L}_i$). The blurring operator A is nonsymmetric; it is determined by Gaussian PSFs defined by the parameters `band` and $\sigma_1 = 4$, $\sigma_2 = 1$. The noise level is $\nu = 5 \cdot 10^{-2}$.

		L_i		P_i		$\widehat{L}_i$	
ℓ	`band`	PSNR	# iter	PSNR	# iter	PSNR	# iter
1	5	30.51	3	-	-	-	-
2	5	31.34	3 2	32.41	3 2	31.96	2 2
3	5	31.50	3 2 2	32.13	3 2 2	31.96	3 2 2
1	7	28.74	3	-	-	-	-
2	7	29.42	2 2	30.21	2 2	29.86	2 2
3	7	29.87	3 2 2	30.23	3 2 2	30.16	3 2 2
1	9	28.07	4	-	-	-	-
2	9	27.88	2 2	28.31	2 2	28.70	2 3
3	9	26.53	1 2 3	29.13	1 2 2	28.70	1 2 2

TABLE 5.3

Example 5.2. This table differs from Table 5.2 only in that LSQR-based cascadic multilevel methods are applied instead of RRGMRES-based multilevel methods.

		L_i		P_i	
ℓ	`band`	PSNR	# iter	PSNR	# iter
2	5	30.81	2 2	31.54	2 2
3	5	31.16	3 2 2	31.59	3 2 2
2	7	29.20	2 2	29.89	2 2
3	7	29.49	3 2 2	29.92	3 2 2
2	9	28.33	2 3	28.71	2 3
3	9	28.51	1 1 3	29.01	1 1 2

TABLE 5.4

Example 5.2. Restoration of corrupted versions of the `corner` image by LSQR-based cascadic multilevel methods, using the averaging restriction operators for obtaining the vectors b_i^δ in (2.7). The table reports PSNR-values of the restored images and the number of iterations (# iter) required both for piecewise linear prolongation operators (L_i) and nonlinear prolongation operators (P_i). The images to be restored are the same as for Table 5.2.

		L_i		P_i	
ℓ	band	PSNR	# iter	PSNR	# iter
1	5	24.85	3	-	-
2	5	25.03	4 2	26.93	4 3
3	5	24.89	3 3 2	26.86	3 3 3
1	7	23.00	3	-	-
2	7	24.84	3 3	24.90	3 3
3	7	33.51	3 3 2	24.92	3 3 3
1	9	22.91	4	-	-
2	9	23.50	4 3	23.52	4 3
3	9	22.18	1 3 3	23.62	1 4 3

TABLE 5.5

Example 5.2. Restoration of corrupted versions of the `lizard` image by RRGMRES-based cascadic multilevel methods. The table reports PSNR-values of the restored images and the number of iterations (# iter) required for piecewise linear prolongation operators (L_i) and nonlinear prolongation operators (P_i). The blurring operator A is nonsymmetric; it is determined by Gaussian PSFs defined by the parameters `band` and $\sigma_1 = 4$, $\sigma_2 = 1$. The noise level is $\nu = 5 \cdot 10^{-2}$.

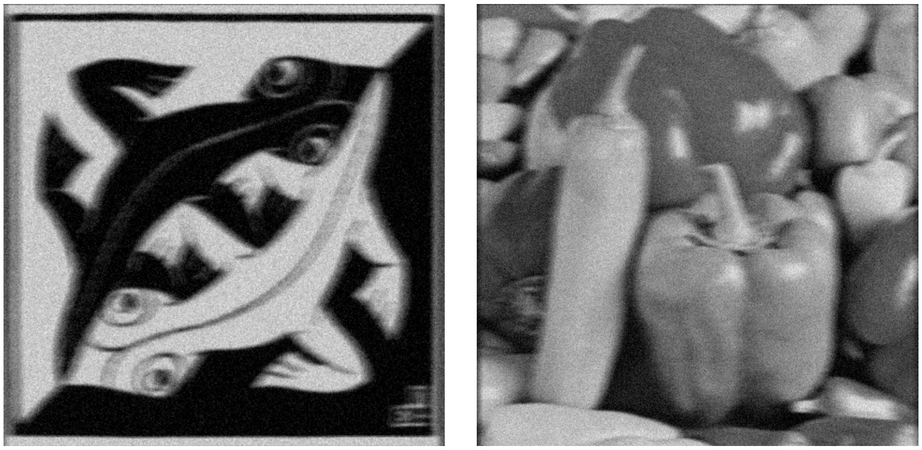
		L_i		P_i	
ℓ	band	PSNR	# iter	PSNR	# iter
1	5	27.77	9	-	-
2	5	27.99	11 7	28.64	11 7
3	5	27.98	3 11 7	28.78	3 11 7
1	7	26.07	10	-	-
2	7	26.85	6 9	27.12	6 9
3	7	26.91	3 6 9	27.43	3 6 9
1	9	24.91	12	-	-
2	9	25.59	10 10	25.97	10 10
3	9	25.30	10 15 10	25.98	10 12 10

TABLE 5.6

Example 5.2. This table differs from Table 5.4 only in that LSQR-based cascadic multilevel methods are applied instead of RRGMRES-based multilevel methods.

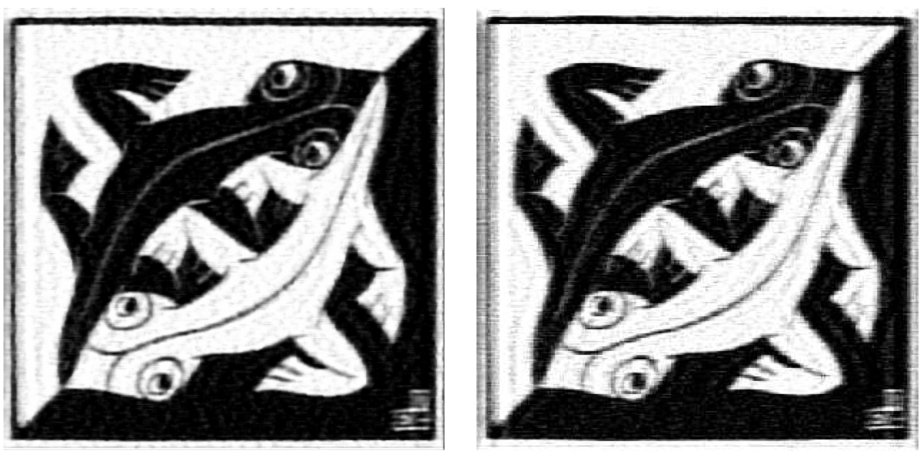
ing in the last step of the algorithm. The computations reported in the columns labeled "L_i" and "$\widehat{L}_i$" are covered by Theorem 2.4, while the results reported in the columns labeled "P_i" are not. The tables show cascadic multilevel methods to determine restorations of higher quality than 1-level methods, and nonlinear prolongation followed by smoothing to yield better restorations than linear prolongation. The restoration quality is measured in terms of PSNR-values. The tables also show the linearized prolongation operators $\widehat{L}_i$ to produce restorations with almost the same PSNR-values as the non-linear prolongation operators P_i with smoothing.

We also compare the weighted least-squares restriction operators in (2.7) with the averaging restriction operators used in [20]. The averaging restriction operator from W_i to W_{i-1} replaces groups of four adjacent pixels in the image represented by b_i^δ by one pixel, whose value is the average of the values of the four pixels it replaces. Table 5.4 reports results obtained with an LSQR-based cascadic multilevel method using averaging restriction operators. Comparison of Tables 5.3 and 5.4 shows that the PSNR-values reported in the former are higher. This depends on that the weighted least-squares restriction operators (2.7) handle image discontinuities better. Finally,



(a) (b)

Fig. 5.4. *Example 5.3.* `lizard` *and* `pepper` *images perturbed by motion blur with* `width`$= 15$, $\theta = 10$, *and noise-level* $\nu = 1 \cdot 10^{-1}$.



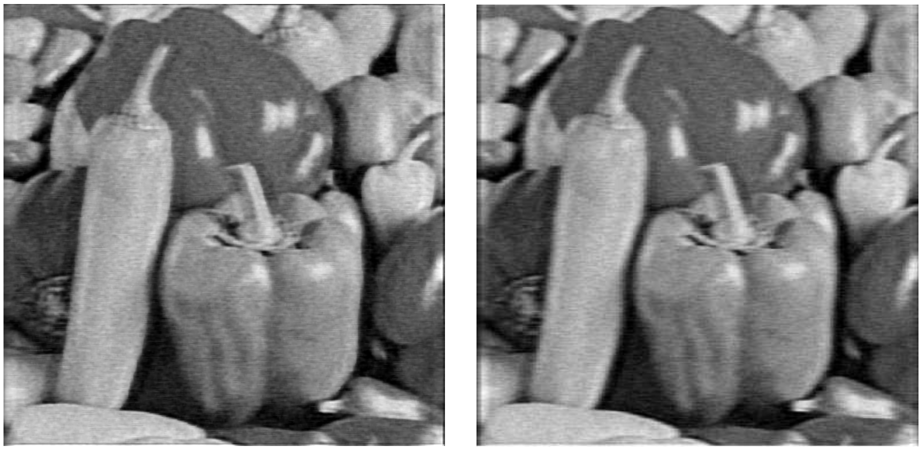
(a) (b)

Fig. 5.5. *Example 5.3. (a) Restoration of the corrupted* `lizard` *image determined by the* 3*-level LSQR-based multilevel method, and (b) restoration obtained by* 1*-level LSQR.*

Tables 5.5-5.6 display results for the `lizard` image; they are analogous to Tables 5.2-5.3.

The arithmetic work required by the cascadic multilevel methods is dominated by the matrix-vector product evaluations on the finest level. Their number therefore can be used to estimate the arithmetic effort required by the cascadic multilevel methods. The tables of this example show the number of matrix-vector product evaluations on the finest level often to be smaller for ℓ-level methods, with $\ell > 1$, than for the corresponding 1-level method. However, the significant advantage of cascadic multilevel methods over the corresponding 1-level iterative methods is the higher quality of the restorations determined. □

Example 5.3. We apply Algorithm 2.2 to the restoration of the images shown in Fig.5.1(b) and (c) that have been contaminated by linear motion blur and noise. The



(a) (b)

FIG. 5.6. *Example 5.3. (a) Restoration of the corrupted* `pepper` *image determined by the* 3*-level LSQR-based multilevel method, and (b) restoration obtained by* 1*-level LSQR.*

	RRGMRES		LSQR	
ℓ	PSNR	# iter	PSNR	# iter
1	22.35	1	25.11	3
2	24.91	2 1	25.82	10 2
3	24.49	2 1 1	26.11	13 11 2

TABLE 5.7
Example 5.3. Restoration of the blurred and noisy `lizard` *image shown in Figure 5.4(a). The table shows PSNR-values of the restored images and the number of iterations (# iter) required by RRGMRES- and LSQR-based cascadic multilevel methods.*

	RRGMRES		LSQR	
ℓ	PSNR	# iter	PSNR	# iter
1	15.21	2	17.25	4
2	16.04	2 1	18.82	14 3
3	17.08	4 2 1	19.29	11 10 3

TABLE 5.8
Example 5.3. Restoration of the blurred and noisy `pepper` *image shown in Figure 5.4(b). The table shows PSNR-values of the restored images and the number of iterations (# iter) required by RRGMRES- and LSQR-based cascadic multilevel methods.*

PSF is represented by a line segment of length r pixels in the direction of the motion. The angle θ (in degrees) specifies the direction; it is measured counter-clockwise from the positive x-axis. The PSF takes on the value r^{-1} on this segment and vanishes elsewhere. We refer to the parameter r as the `width`. The larger the `width`, the more ill-conditioned the matrix A, and the more difficult the restoration task. Figure 5.4 displays `lizard` and `pepper` images that have been contaminated by motion blur defined by `width`$= 15$ and $\theta = 10$, and by noise of level $\nu = 1 \cdot 10^{-1}$. The `lizard` image is representative of black and white images with well-defined edges, while the `pepper` image is a gray-scale photograph with smoothed edges.

Figure 5.5(a) shows the restoration of the corrupted `lizard` image of Figure 5.4(a)

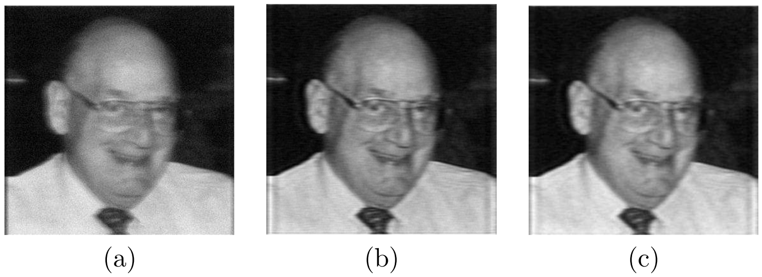


FIG. 5.7. *Example 5.4. (a) Image perturbed by motion blur and noise of level $\nu = 5 \cdot 10^{-2}$. (b) Restoration determined by 1-level Algorithm 2.2, (c) Restoration determined by Algorithm 2.2 with 2 levels using LSQR.*

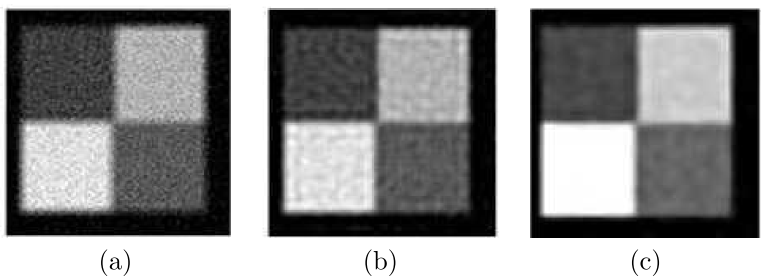


FIG. 5.8. *Example 5.5. (a) Image perturbed by Gaussian nonsymmetric blur and noise of level $\nu = 1 \cdot 10^{-1}$. (b) Restoration determined by the nonlinear PDE-model (1.5) using a semi-implicit discretization. (c) Restoration determined by Algorithm 2.2 with 2 levels using LSQR.*

determined by Algorithm 2.2 based on LSQR, using 3 levels, nonlinear prolongation, and smoothing. The image obtained by 1-level LSQR is shown in Figure 5.5(b). Visual comparison shows the 3-level LSQR-based cascadic multilevel method to give the most pleasing restoration. This is in agreement with the PSNR-values reported in Table 5.7. Analogously, Figure 5.6(a) provides a visual comparison of restorations of the `pepper` image determined by the 3-level LSQR-based cascadic multilevel method with nonlinear prolongation operators and smoothing. The restoration determined by 1-level LSQR is shown by Figure 5.6(b). Table 5.8 compares the PSNR-values for computed restorations of the `pepper` image. The table confirms our earlier observation that LSQR-based cascadic multilevel methods determine better restorations than RRGMRES-based cascadic multilevel methods when the noise-level of the contaminated image is high. □

Example 5.4. We determine restorations of a contaminated version of the original 360×360-pixel `Varga` image displayed in Figure 5.1(d). The image is degraded by linear motion blur defined by `width` $= 15$ and $\theta = 10$, and by noise of level $\nu = 5 \cdot 10^{-2}$. Figure 5.7(a) displays the contaminated image. The restoration obtained by 1-level LSQR is shown in Figure 5.7(b). The PSNR-value for the restored image is 27.40. This method requires 4 iterations to satisfy the discrepancy principle. Since the available image has low resolution, we apply Algorithm 2.2 with only 2 levels. The algorithm carries out 1 and 3 LSQR-iterations on the coarse and fine levels, respectively, and yields a restored image with PSNR=28.32. The latter image is shown in Figure 5.7(c). □

Example 5.5. This example illustrates the performance of the non-linear approach based on equations (1.5) with Perona-Malik diffusivity (1.7) to image restoration. Space-discretization is carried out by finite volumes and gives (1.11). Time-discretization is achieved by a semi-implicit time stepping scheme, in which the u-value from the previous time step, $u^{\ell-1}$, is used in the evaluation of $L(u)$ at the present time step; see [13] for details. At step ℓ we have to solve the linear system of equations

$$[I - \tau(\alpha\, L(u^{\ell-1}) - A^T A)]u^{\ell} = u^{\ell-1} + \tau A^T b^{\delta}, \tag{5.3}$$

for u^{ℓ}.

Semi-implicit time discretization of (1.11) allows larger time steps τ than explicit methods. Difficulties with semi-implicit schemes include that the matrix in the left-hand side of (5.3) typically cannot be factored or inverted inexpensively. Moreover, due to the blur contribution, there still is an upper bound for the size of the time steps that secures stability, though this bound is less restrictive than for explicit integration methods. Nevertheless, this bound in combination with the high computational cost of solving (5.3) at every time step makes semi-implicit solution methods expensive.

The 100×100-pixel image to be recovered is a detail of a blurred and noisy version of the `corner` image. The blur is nonsymmetric and Gaussian, determined by the parameters $\sigma_1 = 4$, $\sigma_2 = 4.5$, and $\texttt{band} = 5$. The additive white Gaussian noise is of level $\nu = 1 \cdot 10^{-1}$. Figure 5.8(a) displays the contaminated image.

Five time steps of size $\tau = 1 \cdot 10^{-4}$ with the semi-implicit scheme yields the image shown in Figure 5.8(b). The time step seems small because of the choice of the space discretization step which is normalized to $1/\sqrt{n}$ in the computation. The time step is chosen as a compromise between stability conditions and computational effort.

We only carried out 5 time steps because more steps do not produce a significantly improved restoration. The computation of this solution requires the evaluation of 10 matrix-vector products with A or A^T. However, the major computational effort is the solution of the 5 linear systems of equations (5.3). We remark that the blur causes these linear systems of equations to be very ill-conditioned.

Figure 5.8(c) displays a restoration determined by the 3-level LSQR-based multilevel method. The latter method requires the evaluation of 6 matrix-vector products with A or A^T.

PSNR-values for the restored images furnish a quantitative comparison. The image restored by the nonlinear PDE model has PSNR = 20.86, while the image determined by the multilevel method has PSNR = 25.86, a significantly larger value.

The fairly poor performance of the nonlinear model (1.5) may depend on that there are two regularization parameters, α and T, the length of the time interval of integration. The optimal values of these parameters are difficult to determine for general non-linear evolution PDE-models. □

Example 5.6. We have in all the above examples assumed the availability of a fairly accurate estimate of the noise-level ν. This example illustrates that the Perona-Malik differential equation can be applied to estimate the amount of noise in an available image b^{δ}. This estimate then can be used in place of the parameter δ in the Stopping Rules 2.1 and 2.3.

Let b^{δ} denote the available noise-contaminated right-hand side, and let $\hat{b}$ be the associated noise-free vector. We integrate the discretized Perona-Malik nonlinear diffusion equation (4.2) for 50 time steps of size 0.2 with initial function $u^0 = b^{\delta}$. This yields the partially denoised vector b'. Table 5.9 displays the exact discrepancy $\delta = \|b^{\delta} - \hat{b}\|$ and the estimated discrepancy $\delta' = \|b' - b^{\delta}\|$. The table suggests that

ν	δ	PSNR	# iter	δ'	PSNR	# iter
$1 \cdot 10^{-2}$	1.01	33.96	3 4 7	1.29	32.06	3 4 3
$5 \cdot 10^{-2}$	5.06	30.23	3 2 2	5.17	30.40	3 2 2
$1 \cdot 10^{-1}$	10.13	29.21	3 1 2	10.16	29.16	3 1 2
$5 \cdot 10^{-1}$	50.67	24.33	3 1 1	48.88	24.45	3 1 1

TABLE 5.9

Example 5.6. Estimation of discrepancies for blurred and noisy `corner` *images. The exact discrepancies are denoted by δ. Estimates of the discrepancies determined by integrating the discretized Perona-Malik diffusion equation are denoted by δ'. The table shows PSNR-values of the restored images and the number of iterations (# iter) required by LSQR-based cascadic multilevel methods.*

integrating the Perona-Malik diffusion equation by an explicit method can provide a useful estimate of the discrepancy. The computational work required is insignificant, when compared to the computational effort required for restoration.

The partially denoised image b' can be used, instead of b^δ, as initial available image in Algorithm 2.2. We applied this approach to restore a `corner` image, which has been contaminated by nonsymmetric Gaussian blur, determined by the parameters `band` $= 7$, $\sigma_1 = 4$, $\sigma_2 = 1$, as well as by noise corresponding to several noise-levels ν. The PSNR-values of the restored images determined by the 3-level LSQR-based multilevel method, as defined by Algorithm 2.2, are reported in column six of Table 5.9; the associated numbers of iterations are shown in column seven. Columns four and five of Table 5.9 display the results obtained with the 3-level LSQR-based Algorithm 2.2 when using the initial image b^δ and the exact noise level δ in the Stopping Rules 2.1 and 2.3. Table 5.9 leads us to conclude that we can apply Algorithm 2.2 successfully as described, also in situations when no accurate estimate of the norm of the noise in the available image b^δ is explicitly known. $\Box$

6. Conclusion. Several methods for the restoration of images that have been contaminated by nonsymmetric blur and noise are compared. Visual inspection of the images shown in Section 5, as well as computed PSNR-values, show cascadic multilevel methods with nonlinear edge-preserving and noise-reducing prolongation operators to give more accurate restorations than 1-level methods applied on the finest level only. We also illustrate that multilevel methods can give restorations of higher accuracy than fully nonlinear models (1.5), and demand less computational effort.

The multilevel methods described require that an estimate of the norm of the noise in the contaminated image be available. We discuss how such an estimate can be computed by integrating the Perona-Malik differential equation for a few time steps. The possibility of computing a fairly accurate estimate of the norm of the noise makes it possible to apply Algorithm 2.2 also to restoration problems for which no such estimate is explicitly known.

Acknowledgments. This work is supported by a MIUR-Cofin 2006 project grant, by the University of Bologna “Funds for Selected Research Topics,” and by an OBR Challenge grant. LR would like to thank Fiorella Sgallari for enjoyable visits to Bologna, during which work for this paper was carried out.

REFERENCES

[1] Å. Björck, Numerical Methods for Least Squares Problems, SIAM, Philadelphia, 1996.
[2] P. Brianzi, P. Favati, O. Menchi, and F. Romani, A framework for studying the regularizing properties of Krylov subspace methods, Inverse Problems, 22 (2006), pp. 1007–1021.

[3] A. Buades, B. Coll, and J. M. Morel, The staircasing effect in neighborhood filters and its solution, IEEE Trans. Image Processing, 15 (2006), pp. 1499–1505.
[4] D. Calvetti, B. Lewis, and L. Reichel, Restoration of images with spatially variant blur by the GMRES method, in Advanced Signal Processing Algorithms, Architectures, and Implementations X, ed. F. T. Luk, Proceedings of the Society of Photo-Optical Instrumentation Engineers (SPIE), vol. 4116, The International Society for Optical Engineering, Bellingham, WA, 2000, pp. 364–374.
[5] D. Calvetti, B. Lewis, and L. Reichel, On the choice of subspace for iterative methods for linear discrete ill-posed problems, Int. J. Appl. Math. Comput. Sci., 11 (2001), pp. 1069–1092.
[6] D. Calvetti, B. Lewis, and L. Reichel, On the regularizing properties of the GMRES method, Numer. Math., 91 (2002), pp. 605–625.
[7] D. Calvetti, B. Lewis, and L. Reichel, GMRES, L-curves and discrete ill-posed problems, BIT, 42 (2002), pp. 44–65.
[8] T. F. Chan and J. Shen, Image Processing and Analysis: Variational, PDE, Wavelet, and Stochastic Methods, SIAM, Philadelphia, 2005.
[9] M. Donatelli and S. Serra-Capizzano, On the regularization power of multigrid-type algorithms, SIAM J. Sci. Comput., 27 (2006), pp. 2053–2076.
[10] M. Donatelli and S. Serra-Capizzano, Filter factor analysis of an iterative multilevel regularizing method, Electron. Trans. Numer. Anal., 29 (2008), pp. 163–177.
[11] H. W. Engl, M. Hanke, and A. Neubauer, Regularization of Inverse Problems, Kluwer, Dordrecht, 1996.
[12] M. Gulliksson and P.-Å. Wedin, Modifying the QR decomposition to constrained and weighted least-squares, SIAM J. Matrix Anal., 13 (1992), pp. 1298–1313.
[13] A. Handlovicová, K. Mikula, and F. Sgallari, Semi-implicit complementary volume scheme for solving level set like equations in image processing and curve evolution, Numer. Math., 93 (2003), pp. 675–695.
[14] M. Hanke, Conjugate Gradient Type Methods for Ill-Posed Problems, Longman, Essex, 1995.
[15] P. C. Hansen, J. G. Nagy, and D. P. O'Leary, Deblurring Images: Matrices, Spectra, and Filtering, SIAM, Philadelphia, 2006.
[16] T. Huckle and J. Staudacher, Multigrid preconditioning and Toeplitz matrices, Electron. Trans. Numer. Anal., 13 (2002), pp. 81–105.
[17] M. Jacobsen, P. C. Hansen, and M. A. Saunders, Subspace generalized LSQR for discrete ill-posed problems, BIT, 43 (2003), pp. 975–989.
[18] T. K. Jensen and P. C. Hansen, Iterative regularization with minimum-residual methods, BIT, 47 (2007), pp. 103–120.
[19] B. Lewis and L. Reichel, Arnoldi-Tikhonov regularization methods, J. Comput. Appl. Math., 226 (2009), pp. 92–102.
[20] S. Morigi, L. Reichel, F. Sgallari, and A. Shyshkov, Cascadic multiresolution methods for image deblurring, SIAM J. Imaging Sci., 1 (2008), pp. 51–74.
[21] A. S. Nemirovskii, The regularization properties of the adjoint gradient method in ill-posed problems, USSR Comput. Math. Math. Phys, 26 (2) (1986), pp. 7–16.
[22] M. K. Ng, R. H. Chan, and W.-C. Tang, A fast algorithm for deblurring models with Neumann boundary conditions, SIAM J. Sci. Comput., 21 (1999), pp. 851–866.
[23] C. C. Paige and M. A. Saunders, LSQR: An algorithm for sparse linear equations and sparse least squares, ACM Trans. Math. Software, 8 (1982), pp. 43–71.
[24] P. Perona and J. Malik, Scale-space and edge detection using anisotropic diffusion, IEEE Trans. Pattern Anal. Mach. Intell., 12 (1990), pp. 629–639.
[25] L. Reichel and A. Shyshkov, Cascadic multilevel methods for ill-posed problems, J. Comput. Appl. Math., in press.
[26] L. Reichel and Q. Ye, Breakdown-free GMRES for singular systems, SIAM J. Matrix Anal. Appl., 26 (2005), pp. 1001–1021.
[27] L. Rudin, S. Osher, and E. Fatemi, Nonlinear total variation based noise removal algorithms, Physica D, 60 (1992), pp. 259–268.
[28] Y. Saad, Iterative Methods for Sparse Linear Systems, 2nd ed., SIAM, Philadelphia, 2003.
[29] R. S. Varga, Matrix Iterative Analysis, 2nd ed., Springer, Berlin, 2000.
[30] J. Weickert, B. M. H. Romeny, and M. A. Viergever, Efficient and reliable schemes for nonlinear diffusion filtering, IEEE Trans. Image Process., 7 (1998), pp. 398-410.
[31] M. Welk, D. Theis, T. Brox, and J. Weickert, PDE-based deconvolution with forward-backward diffusivities and diffusion tensors, in Scale-Space and PDE Methods in Computer Vision, eds. R. Kimmel, N. Sochen, and J. Weickert, Lecture Notes in Computer Science, vol. 3459, Springer, Berlin, 2005, pp. 585–597.
[32] J. Zhu and Y. M. Chen, Multilevel grid method for history matching multi-dimensional multi-

phase reservoir models, Appl. Numer. Math., 10 (1992), pp. 159–174.