ITERATIVE PARAMETER-CHOICE & MULTIGRID METHODS FOR ANISOTROPIC DIFFUSION DENOISING

D. CHEN[†], S. MACLACHLAN[†], AND M. KILMER[†]

Abstract. Anisotropic diffusion methods are well-known to give good qualitative results for image denoising. This paper gives a review of the anisotropic diffusion methodology and its application to image denoising. We propose a fixed-point iteration using a multigrid solver to solve a regularized anisotropic diffusion equation, which is not only well-posed, but also has a nontrivial steady-state solution. A new regularization parameter-choice method (Brent-NCP), combining Brent's method and the normalized cumulative periodogram information of the misfit, is also introduced. We test our algorithm on several common test images with different noise levels. The experimental results demonstrate the effectiveness of the anisotropic diffusion with multigrid approach and the broad applicability of the Brent-NCP parameter-choice algorithm.

Key words. anisotropic diffusion, robust multigrid, normalized cumulative periodogram, Brent's method, regularization, parameter choice

AMS subject classifications. 65F10, 65N55, 35K55

1. Introduction.

It is well known that during the formation, transmission, and recording processes, images deteriorate with various types of noise. Therefore, it is important to eliminate the noise efficiently and automatically. Many image denoising techniques have been proposed over the years, and a good review of them can be found in [8]. In particular, with developments in computer technology, methods based on partial differential equations (PDEs) have been extensively studied as approaches to image denoising.

Anisotropic diffusion (AD), first introduced by Perona and Malik [27], has been widely accepted as a method for removing noise while preserving and enhancing edges. Many papers have proposed different techniques for solving the anisotropic diffusion equations [4]. Among them, the additive operator splitting (AOS) scheme suggested by Weickert [42] is very impressive, and has the advantages that it is stable and easy to implement.

Another option for the efficient solution of these systems is the use of multigrid techniques that are applicable in more general settings, see [7, 35]. The optimality of multigrid methods suggests that they are potentially good solvers for anisotropic diffusion problems. The black box multigrid (BoxMG) technique, first introduced in [2, 14], uses geometrically structured coarse grids in combination with an interpolation operator designed to account for the effects of discontinuous diffusivities to achieve fast multigrid convergence in many situations. This makes BoxMG a potential fast solver for anisotropic diffusion denoising because of the inherently structured nature of digital images. Another possible choice is to use algebraic multigrid methods (AMG). First proposed in [6], AMG is designed to utilize classical multigrid principles to obtain fast solution for a wide range of problems. It is robust and effective especially when the coefficients of the PDE are discontinuous and vary widely [31, 35]. While there has been some previous research using multigrid methods to solve anisotropic diffusion equations [1, 12, 21, 16], there are few comparisons between BoxMG and AMG within the image processing literature [26].

[†] Department of Mathematics, Tufts University, 503 Boston Avenue, Medford, MA 02155 (donghui.chen@tufts.edu, misha.kilmer@tufts.edu, scott.maclachlan@tufts.edu). The work of DC and SM was sponsored by the National Science Foundation under grant DMS-0811022.

Besides finding a fast algorithm to solve linearized anisotropic diffusion equations, choosing a good regularization parameter is also critical in anisotropic diffusion denoising. Since previous methods, such as Weickert's relative variance method [40], Mrázek's decorrelation method [23], etc., do not yield optimal regularization parameters for anisotropic denoising, we propose a new inexpensive parameter-choice method based on Brent's method and the normalized cumulative periodogram (NCP) of the misfit vector [19, 32] in this paper. Numerical experiments show that the proposed algorithm can find near-optimal regularization parameters for the anisotropic diffusion process quickly and efficiently.

Our approach differs from previous work in the following two aspects:

1. We consider a different objective diffusion equation, which is not only well-posed, but also has a nontrivial steady-state solution. In practice, by searching for the steady-state solution directly, the proposed solver converges very quickly.
2. By introducing the new Brent-NCP parameter-choice method, the proposed denoising algorithm chooses the regularization parameters automatically. This makes our algorithm more adaptive and efficient. Furthermore, the application of the parameter-choice algorithm is not limited to anisotropic diffusion based denoising. It can, in fact, be applied to a very broad range of parameter-dependent image restoration algorithms.

The remainder of this paper is organized as follows. Section 2 presents a review of the anisotropic diffusion equation in image denoising and introduces our cost functional. A semi-implicit discretization technique is also discussed here. Section 3 introduces the new Brent-NCP regularization parameter-choice algorithm. Section 4 presents experimental results for the new approach and comparisons to other common approaches from the image denoising literature. The conclusions of this paper are presented in Section 5.

2. Anisotropic Diffusion Review.

White noise is one of the most common problems in image processing. Intuitively, we can smooth out the noise by convolving the image with a Gaussian kernel, which is equivalent to solving the heat equation. However, convolution with a linear Gaussian filter not only removes high-frequency noise but also blurs edges and destroys finer textures. Anisotropic diffusion denoising is based on the idea of applying a smoothing process that depends on local properties of the image, see [4, 38].

2.1. Perona-Malik Diffusion Model.

In [27], Perona and Malik introduced the anisotropic diffusion approach to replace classical isotropic diffusion in image denoising. This approach can avoid the excessive smoothing effect that occurs with isotropic diffusion procedures, such as Gaussian filters. On the continuous domain, the anisotropic diffusion equation for an image, I, is given by

$$\left\{ \begin{array}{l} \frac{\partial I}{\partial t} = \operatorname{div}(c(|\nabla I|^2)\nabla I) \quad \text{in} \quad \Omega \times (0,T) \\ \frac{\partial I}{\partial N} = 0 \quad \text{on} \quad \partial\Omega \times (0,T) \\ I(0,x) = I_0(x) \quad \text{in} \quad \Omega \end{array} \right. \tag{2.1}$$

where ∇ is the gradient operator, div is the divergence operator, and

$$c(s^2) : [0,\infty] \to [0,\infty]$$

describes the diffusivity. The differential equation has initial condition $I_0(x)$, which is the noisy image.

Defining the flux function

$$\Phi(s) := c(s^2)s,$$

it is shown in [4] that the blurring-enhancing process depends on the sign of derivative of the flux function,

$$b(s^2) := \Phi'(s) = c(s^2) + 2s^2c'(s^2).$$

If $b(s^2) > 0$, the Perona-Malik (PM) model is a forward parabolic equation, and all texture is blurred; while if $b(s^2) < 0$, the PM model is a backward parabolic equation, and the texture is sharpened. Given a threshold, K, the PM approach for choosing $c(s^2)$ shows the desirable result of blurring small discontinuities in an image, I, in regions where $|\nabla I| < K$, and sharpening edges in an image, I, in regions where $|\nabla I| \geq K$. This gives rise to the following assumptions on $c(s^2)$,

$$\begin{cases} c(s^2) : [0,\infty] \to [0,\infty] \quad \text{decreasing}, \\ c(0) = 1, \\ b(s^2) = c(s^2) + 2s^2c'(s^2) < 0 \quad \text{for} \quad s \geq K. \end{cases} \tag{2.2}$$

In their original paper, Perona and Malik choose the diffusivity to be $c(s^2) = \frac{1}{1+s^2/K^2}$, where K is a threshold determined by the noise level, see [9, 28]. This diffusivity is a canonical example satisfying the assumptions in (2.2). In this paper, we use the same diffusivity as the one used in Perona and Malik's paper.

While numerical results with the PM model (2.1) are quite impressive, the forward-backward diffusion process itself is not well-posed. This is the so-called Perona-Malik paradox. In [20], Kichenassamy proves that if the initial image, $I_0(x)$, is not infinitely differentiable, there is no weak solution of (2.1). Consequently, the notion of a "generalized solution", which is piecewise linear and contains jumps, is introduced. However, one should neither expect uniqueness nor stability with respect to the initial image. Examples of significantly differing solutions with nearly identical initial data have been reported [20].

2.2. Regularization. Although the ill-posedness of the PM model can be handled by applying an implicit spatial discretization [41], in order to make the numerical implementation more predictable, it is more natural to introduce regularization into the continuous PM model. Catté et al. [10] introduce a spatial regularization that makes the forward-backward diffusion process become well-posed. The idea is to use a smoothed version, $G_\sigma * \nabla I$, of the image gradient ∇I in the diffusivity $c(|\nabla I|^2)$. Here, G_σ can be any "low-pass filter". In this paper, we assume that G_σ is a Gaussian kernel with standard deviation $\sigma = 0.5$. Since $G_\sigma * \nabla I = \nabla(G_\sigma * I)$, the spatially regularized PM model becomes

$$\frac{\partial I}{\partial t} = \operatorname{div}(c(|\nabla(G_\sigma * I)|^2)\nabla I) \tag{2.3}$$

Catté et al. [10] proved that there exists a unique solution for the regularized PM equation (2.3) with corresponding initial and boundary values. Furthermore, this spatial regularization makes the filter insensitive to noise. This avoids the shortcoming of the original PM model, which cannot distinguish between "true" edges and "false" edges created by the noise. Weickert et al. [42] proposed an additive operator splitting (AOS) scheme to solve (2.3). The diffusion stopping time, T, is the parameter controlling the restored image quality.

While this spatial regularization makes the PM model well-posed, it leads to a process where the solution always converges to a constant steady-state solution [39]. In order to get a nontrivial result, it is then required to specify a stopping time, T_0, such that the restored image, $I(T_0)$, is a good representation of the denoised image. Sometimes, it is attempted to circumvent this task by adding an additional reaction term [24],

$$\frac{\partial I}{\partial t} = \mathrm{div}(c(|\nabla I|^2)\nabla I) + \lambda(I_0 - I). \tag{2.4}$$

The reaction term, $(I_0 - I)$, keeps the steady-state solution close to the original image, I_0. In practice, such a modification shifts the problem of specifying a stopping time, T_0, to the problem of determining the non-negative regularization parameter, λ.

Combining the spatial regularization (2.3) and reaction anisotropic diffusion (2.4) approaches, we get the anisotropic diffusion equation

$$\frac{\partial I}{\partial t} = \mathrm{div}(c(|\nabla(G_\sigma * I)|^2)\nabla I) + \lambda(I_0 - I), \tag{2.5}$$

where we assume that G_σ is fixed and known. As discussed above, this PDE is not only well-posed, but also has a non-trivial steady-state solution satisfying

$$0 = \mathrm{div}(c(|\nabla(G_\sigma * I)|^2)\nabla I) + \lambda(I_0 - I). \tag{2.6}$$

Thus, for a fixed regularization parameter λ, two approaches are possible: one could timestep and lag to solve (2.5) to steady state, or one could solve (2.6) using a fixed point iterative approach. In either case, finding a near optimal regularization parameter λ_{opt} is critical to obtaining a good restored image, and will require that either (2.5) or (2.6) be solved for different values of λ. We discuss methods for choosing such a value in Section 3, and we present a new algorithm for estimating λ_{opt} that can be employed whether one chooses to use (2.5) or (2.6).

In the case of solving (2.5) for a fixed λ, the issues to consider are the time step size and how accurately to solve at each time step. Weickert's AOS scheme is one method that can also be employed to solve (2.5) for each fixed λ. Unlike using AOS to solve (2.3), the stopping time T is not considered a regularization parameter – rather, the stopping time is given by the time step at which the solution appears to have reached steady-state. The case of solving (2.6) by fixed point iteration for a given value of λ, in contrast to the AOS approach for (2.5), requires a more accurate linear solve at each iteration. The potential comparative upside, however, is that fewer overall iterations will needed, and unintentional smoothing that may occur as a result of inaccurate solves during timestepping to solve (2.5) are avoided. In this paper, we consider two solvers for the linear systems that must be solved accurately with the fixed point approach: BoxMG and AMG. Numerical results that compare AOS for solving (2.5) vs. fixed point iteration with BoxMG to solve (2.6) vs. fixed point iteration with AMG to solve (2.6) are presented in Section 4. In all three cases, in order to do a fair comparison, the same regularization parameter selection scheme was employed.

2.3. Discretization. In [43], the authors compared three discretization schemes: both explicit and semi-implicit schemes based on a 3×3 stencil, and an explicit scheme based on a 5×5 stencil. They concludes that the 5×5 stencil explicit discretization scheme is superior than the explicit scheme based on 3×3 stencils in terms of rotation invariance, accuracy, and avoidance of blurring artifacts. However, results for the 3×3 AOS-stabilized semi-implicit approach are comparable to those for the 5×5 explicit stencil. Therefore, we use a semi-implicit discretization technique to discretize (2.6), which retains the memory advantage of using a 3×3 stencil. Instead of using a simple central difference scheme as the authors did in [42], we use more points in computing $c(|\nabla(G_\sigma * I)|^2)$ to increase the stability [4]. Assuming the regularization parameter, λ, is known, the following fixed-point iteration is used to compute the solution of (2.6),

$$\lambda(I^{n+1} - I_0) = \operatorname{div}(c(|\nabla(G_\sigma * I^n)|^2)\nabla I^{n+1}), \tag{2.7}$$

where the superscript, n, denotes a numerical approximation taken at the nth iteration. Writing the divergence term as

$$\operatorname{div}(c\nabla I) = \frac{\partial}{\partial x}\left(c\frac{\partial I}{\partial x}\right) + \frac{\partial}{\partial y}\left(c\frac{\partial I}{\partial y}\right), \tag{2.8}$$

we use central differences to approximate the derivatives of the image, I. In digital images, the distance between adjacent grid points, h, is constant. For simplicity, we omit the distance h in the following discretization formulas. The value of the divergence operator at grid point (i, j) can then be written as

$$\begin{aligned}
\operatorname{div}(c\nabla I)\mid_{i,j} &= c_{i+\frac{1}{2},j}(I_{i+1,j} - I_{i,j}) - c_{i-\frac{1}{2},j}(I_{i,j} - I_{i-1,j}) \\
&\quad + c_{i,j+\frac{1}{2}}(I_{i,j+1} - I_{i,j}) - c_{i,j-\frac{1}{2}}(I_{i,j} - I_{i,j-1}) \\
&= c_{i+\frac{1}{2},j}I_{i+1,j} + c_{i-\frac{1}{2},j}I_{i-1,j} + c_{i,j+\frac{1}{2}}I_{i,j+1} + c_{i,j-\frac{1}{2}}I_{i,j-1} \\
&\quad -(c_{i+\frac{1}{2},j} + c_{i-\frac{1}{2},j} + c_{i,j+\frac{1}{2}} + c_{i,j-\frac{1}{2}})I_{i,j}
\end{aligned}$$

Notice that interpolation is needed to evaluate the diffusivity, $c = c(|\nabla\hat{I}|^2)$, at locations $(i \pm \frac{1}{2}, j)$ and $(i, j \pm \frac{1}{2})$. This can be done as follows, see also Figure 2.1. Denoting $\hat{I} = G_\sigma * I^n$, we use central differences and linear interpolation with a compact stencil to compute the diffusivity

$$c_{i,j+\frac{1}{2}} := c\left((\hat{I}_{i,j+1} - \hat{I}_{i,j})^2 + \left(\frac{\hat{I}_{i+1,j+1} - \hat{I}_{i-1,j+1} + \hat{I}_{i+1,j} - \hat{I}_{i-1,j}}{4}\right)^2\right),$$

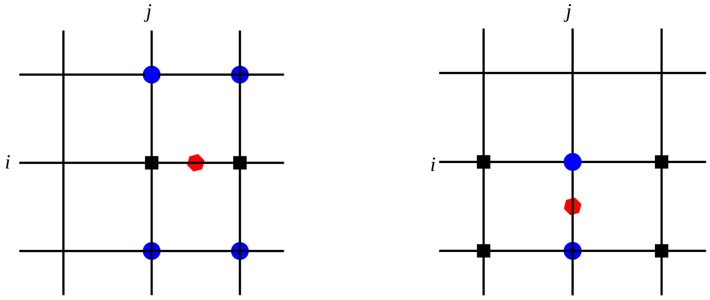
and

$$c_{i+\frac{1}{2},j} := c\left(\left(\frac{\hat{I}_{i+1,j+1} - \hat{I}_{i+1,j-1} + \hat{I}_{i,j+1} - \hat{I}_{i,j-1}}{4}\right)^2 + (\hat{I}_{i+1,j} - \hat{I}_{i,j})^2\right).$$

Because of the Gaussian filter, G_σ, and the average used in computing the diffusivity, c, the above discretization is less sensitive to noise than without the filter. In practice, this also makes the discretization have a rotation-invariance property [4]. On the other hand, using a compact stencil provides a good balance between accuracy and computational time.

Algorithm 1 Fixed-Point Iteration for Fixed λ

```
while $\text{norm}(I^{n+1} - I^n)/\text{norm}(I^n) > \text{tol}_{\text{fp}}$ do
    Compute the matrix $A_n = A(I^n, \lambda)$ in (2.9) with approximation $I^n$
    Compute the solution $I^{n+1}$ of $A_n \text{vec}(I^{n+1}) = \lambda \text{vec}(I_0)$
end while
return $I^{n+1}$
```



(a) $(\nabla \hat{I})_{i,j+\frac{1}{2}}$

(b) $(\nabla \hat{I})_{i+\frac{1}{2},j}$

Fig. 2.1: Grid points involved in the approximation of the diffusivities $c(|\nabla \hat{I}|^2)$ at grid points $(i, j+\frac{1}{2})$ and $(i+\frac{1}{2}, j)$ (marked by hexagons). The grid points represented by dots are used to compute derivatives in the y-direction, while the grid points represented by squares are used to compute derivatives in the x-direction.

The discretization of (2.7) is, then,

$$a_{i,j} I^{n+1}_{i,j} - \left(c_{i+\frac{1}{2},j} I^{n+1}_{i+1,j} + c_{i-\frac{1}{2},j} I^{n+1}_{i-1,j} + c_{i,j+\frac{1}{2}} I^{n+1}_{i,j+1} + c_{i,j-\frac{1}{2}} I^{n+1}_{i,j-1} \right) = \lambda I^0_{i,j},$$

where $I^0 = I_0$, $a_{i,j} = \left(\lambda + \left(c_{i+\frac{1}{2},j} + c_{i-\frac{1}{2},j} + c_{i,j+\frac{1}{2}} + c_{i,j-\frac{1}{2}} \right) \right)$. In matrix-vector notation, the above discrete form can be written as

$$A(I^n, \lambda) \text{vec}(I^{n+1}) = \lambda \text{vec}(I_0), \tag{2.9}$$

where vec(I) denotes the vector obtained by stacking all of the columns of image I into a vector.

2.4. Multigrid solvers. For fixed λ, the linearized diffusion equation (2.9) represents a fixed-point linearization of the nonlinear PDE described by (2.6). This, naturally, leads to the fixed-point iteration given as Algorithm 1. Moreover, the off-diagonal entries of matrix $A(I^n, \lambda)$ are less than or equal to zero, while the diagonal entries are positive, and are strictly diagonally dominant. Thus $A(I^n, \lambda)$ is an M-matrix, and multigrid methods can be effectively used to solve the linearized problem [31, 35].

In general, multigrid solvers for a given discrete matrix make use of a two-part process consisting of a *setup phase* and a *solution phase* [6, 35]. The *setup phase* consists of a recursive coarsening process, defining a series of progressively coarser grids and transfer operators between them. The *solution phase* uses the resulting components to perform normal multigrid cycling, where the relaxation method is fixed to be some simple algorithm, such as Gauss-Seidel, until a desired stopping criterion is satisfied. Normally, a single multigrid V-cycle doesn't solve problem (2.9) exactly; instead, it is applied iteratively. The iteration stops when the relative residual reduction becomes less than a given relative tolerance, tol_{mg}.

Multigrid methods can be split into two categories: geometric multigrid (GMG) methods and algebraic multigrid (AMG) methods. GMG methods use geometrically structured coarse grids within the multigrid process [7]; traditional GMG methods use no information about the matrix, $A(I^n, \lambda)$, in defining the grid-transfer operators and, so, are not robust to the discontinuous diffusivities considered here. However, [2, 14] introduced the black box multigrid (BoxMG) algorithm, which uses geometrically structured coarse grids in combination with an interpolation operator designed to account for the effects of discontinuous diffusivities to achieve fast multigrid convergence in many situations. In contrast to GMG methods that work with structured grids, AMG methods make no assumptions other than that there is a matrix and a right-hand side given. They achieve efficiency by tailoring both the coarse-grid structure and interpolation operator to account for jumps in the coefficients. This makes AMG methods more robust, and they can be applied in a wide variety of applications. However, the cost for this robustness is that AMG methods have more expensive setup and solve processes due to the use of unstructured matrix storage approaches. Full details of the BoxMG and AMG algorithms can be found in [2, 7, 14, 31, 35]. In this paper, we use BoxMG and AMG methods as *black-box* solvers and compare their performance in solving anisotropic diffusion equation (2.6).

3. Choosing Regularization Parameters.

We will be comparing the performance of the AOS scheme for (2.3), (2.5) and fixed-point iterations with multigrid solvers for (2.6). The remaining outstanding issue in solving the anisotropic diffusion equations given in (2.3), (2.5) and (2.6) is the choice of the regularization parameters: the diffusion stopping time, T, and/or the parameter, λ.

3.1. Previous methods.

In [40], Weickert points out that since the variance $var(I_t)$ of the continuum solution of the anisotropic diffusion equation at time t, I_t, is monotonously decreasing, the relative variance

$$\frac{\text{var}(I_t)}{\text{var}(I_0)}, \tag{3.1}$$

decreases monotonically from 1 to 0. This ratio measures the distance of I_t from the initial state I_0 and final state I_∞. Prescribing a certain value for the above ratio provides a criterion for the stopping time. Assuming that the signal-to-noise ratio (SNR) is known, [40] proposes to choose the stopping time, T, to satisfy the relation

$$\frac{\text{var}(I_T)}{\text{var}(I_0)} = \frac{1}{1 + \frac{1}{\text{SNR}}}. \tag{3.2}$$

However, one of the drawbacks of this method is that it requires that the SNR be known for the noisy image. Otherwise, the user must specify a threshold for the ratio (3.1). This shifts the problem of choosing a stopping time to that of choosing a threshold. Moreover, as pointed out by Weickert, criterion (3.2) tends to underestimate the optimal stopping time, as even a well-tuned filter cannot avoid influencing the signal before eliminating the noise. For these reasons, in the numerical results section we will use methods other than (3.2) for selecting the stopping time for AOS.

In [23], Mrázek proposed a decorrelation criterion to choose the diffusion stopping time, which is claimed to outperform (3.2). Given the assumption that the noise is uncorrelated with the unknown true image, the decorrelation method for choosing the diffusion stopping time, T, is to minimize the correlation coefficient (CC),

$$T = \underset{t\geq 0}{\operatorname{argmin}} \frac{\operatorname{cov}(I_t - I_0, I_t)}{\sqrt{\operatorname{var}(I_t - I_0)\cdot \operatorname{var}(I_t)}}.$$

Note that this CC idea can be modified to choose λ in (2.6) according to

$$\lambda = \underset{\lambda}{\operatorname{argmin}} \frac{\operatorname{cov}(I_\lambda - I_0, I_\lambda)}{\sqrt{\operatorname{var}(I_\lambda - I_0)\cdot \operatorname{var}(I_\lambda)}}. \tag{3.3}$$

There are many alternative regularization parameter-choice methods, see [18, 25]; this CC measure itself is not universally accepted within the image processing community [44]. However, an exhaustive comparison of these heuristics is beyond the scope of this paper. The method we propose in the following section provides a viable alternative to these approaches above and is flexible enough that it can be used to find either the stopping time in (2.3) or the λ in (2.6).

3.2. The Brent-NCP parameter-choice method. In [19], Hansen, Kilmer and Kjeldsen proposed a rule that seeks to use all of the information available in the misfit vector, $I - I_0$. The key idea of this method is to choose the regularization parameter for which the misfit vector changes from being dominated by the remaining signal to being like white Gaussian noise. By employing statistical tools, such as the Komolgorov-Smirnov (K-S) test, and fast Fourier transforms, this method leads to a parameter-choice rule based on the normalized cumulative periodogram (NCP), which is particularly well-suited for large-scale problems. As pointed out in [34], the advantage of using the K-S test is that it is sensitive to any difference between the objects. For more details on using cumulative periodograms for regularization parameter selection, see [32] and the references therein.

3.2.1. Normalized cumulative periodogram. Denote the Fourier transform of a 2D $n \times n$ signal X by $\mathcal{F}(X)$. Let $q = \lfloor \frac{n}{2} \rfloor + 1$, where $\lfloor \frac{n}{2} \rfloor$ is the maximum integer less than $\frac{n}{2}$. The *periodogram* of X can be computed as the $q \times q$ matrix P, where the elements of P are given by

$$P_{l,m} = \left|[\mathcal{F}(X)]_{l,m}\right|^2, \quad l, m = 1, 2, \cdots, q.$$

The NCP of X is a useful tool to describe the overall behavior of the periodogram. After reordering the elements of P in order of increasing spatial frequency, $\hat{p} = \operatorname{perm}(\operatorname{vec}(P))$, the NCP of X is defined as a vector, $\operatorname{ncp}(X)$, of length $q^2 - 1$ with elements

$$\operatorname{ncp}(X)_k = \frac{||\hat{p}(2 : k+1)||_1}{||\hat{p}(2 : q^2)||_1}, \quad k = 1, \cdots, q^2 - 1. \tag{3.4}$$

For a signal with only white noise, the expected values of the NCP lie on a straight line, v, between $(0, 0)$ and $(q^2, 1)$. A test of the hypothesis that the signal is white noise can be achieved by constructing two lines parallel to v, i.e. the Kolmogorov-Smirnov limits.

For a given regularization method, let us use λ to represent the regularization parameter. Note that if we are solving (2.3) via the AOS scheme, λ represents a stopping time. Given the computed misfit vector, $r_\lambda = I_0 - I(\lambda)$, if $\text{ncp}(r_\lambda)$ lies within the two limit lines, we accept the hypothesis that the difference between the noisy signal and the restored image is white noise [17]. Hence, the corresponding regularization parameter, λ, is chosen.

In practice, in order to find a near optimal regularization parameter, Hansen et al. suggest to solve the following minimization problem to select the regularization parameter:

$$\underset{\lambda}{\text{argmin}}\, \mathcal{N}(\lambda) := ||v - \text{ncp}(r_\lambda)||_1, \tag{3.5}$$

where r_λ denotes the computed misfit given regularization parameter λ. Numerical experiments (in Section 4) suggest that the minimizer of $\mathcal{N}(\lambda)$ for the regularization methods we consider is unique.

3.3. Brent-NCP algorithm. Efficiently solving the minimization problem (3.5) is essential to the parameter-choice algorithm. Since $\mathcal{N}(\lambda)$ is not obviously differentiable, solving (3.5) when the regularization method is given by (2.6) requires solving (2.6) for different values of λ, which is time consuming. Similarly, when the regularized solution is given by the solution to (2.3) at stopping time $\lambda = T$, if the number of time steps is fixed independent of λ, each new regularization parameter means a new discrete solve of (2.3), and hence solving (3.5) is also quite costly for this regularization approach. Because Brent's method is characterized by quadratic convergence in case of smooth functions and guaranteed linear convergence in case of nonsmooth or oscillatory functions, we propose Algorithm 2 to minimize the number of search steps leading to an efficient optimization process. As the plots in Figures 4.3a and 4.4a illustrate, the shape of $\mathcal{N}(\lambda)$ is such that it justifies the use of this straightforward optimization algorithm which we now describe.

Brent's method combines the golden section search method with a parabolic interpolation method to minimize $\mathcal{N}(\lambda)$ [3, 29]. Starting with 3 different points, on each iteration Brent's method approximates the function using an interpolating parabola through three existing points. The minimum of the parabola is taken as a guess for the minimum of $\mathcal{N}(\lambda)$. If it lies within the bounds of the current interval, then the interpolating point is accepted, and is used to generate a smaller interval. If the interpolating point is not accepted, then the algorithm falls back to a golden section step. The full details of Brent's method, including some additional checks to improve convergence, can be found in [3, 29]. For convenience, we summarize Brent's method applied to (3.5) as Algorithm 2.

In practice, the above Brent-NCP algorithm converges very quickly for all the regularization methods considered. In addition, it can be applied in many situations, not just for minimizing the NCP measure for anisotropic diffusion based denoising. The evaluation of $\mathcal{N}(\lambda)$ in steps 4 and 12 depends on the method of regularization employed. For example, if one is using anisotropic diffusion denoising, evaluation of r_λ requires either the solution to time λ if AOS is employed to solve (2.3) or the solution of (2.6) for the given λ. Total variation is another alternative regularization method (see Section 4.2.3) around which the Brent-NCP parameter selection routine could be wrapped; see Section 4 for details.

4. Experimental Results. This section is devoted to presenting the results obtained with the proposed algorithm. Comparisons with the AOS scheme for (2.3) [42] and (2.5), TV denoising [11, 30] and the block matching 3D (BM3D) method [13] are also presented.

Algorithm 2 Brent-NCP Algorithm

```
Set $a < c$, $b \leftarrow a + \frac{3-\sqrt{5}}{2} \times (c - a)$
$\lambda \leftarrow (a + c)/2$
Compute the misfit $r_\lambda := I_0 - I(\lambda)$, where $I(\lambda)$ is the denoised image given regularization parameter $\lambda$
Compute $\mathcal{N}(\lambda) = ||v - \text{ncp}(r_\lambda)||_1$
while $|\lambda - b| > \text{tol}$ do
    Construct a trial parabolic fit
    if parabolic fit is acceptable then
        Take the parabolic step
    else
        Take a golden section step
    end if
    Update the values $a, b, c, \lambda$, compute $\mathcal{N}(\lambda) = ||v - \text{ncp}(r_\lambda)||_1$.
end while
return $\lambda$
```

4.1. Comparison Measures. After getting the restored image, we compute the mean structure similarity (MSSIM) of the restored and noise-free images as a measurement of the restored image quality [36]. Given any two images, $\mathbf{x}$ and $\mathbf{y}$, the structure similarity (SSIM) measure is defined as

$$\text{SSIM}(\mathbf{x}, \mathbf{y}) = \frac{(2\mu_\mathbf{x}\mu_\mathbf{y} + c_1)(2\text{cov}(\mathbf{x}, \mathbf{y}) + c_2)}{(\mu_\mathbf{x}^2 + \mu_\mathbf{y}^2 + c_1)(\sigma_\mathbf{x}^2 + \sigma_\mathbf{y}^2 + c_2)} \tag{4.1}$$

where $\mu_\mathbf{x}$ and $\mu_\mathbf{y}$ are the means of images $\mathbf{x}$ and $\mathbf{y}$ respectively, $\sigma_\mathbf{x}$ and $\sigma_\mathbf{y}$ are the variances of images $\mathbf{x}$ and $\mathbf{y}$, $\text{cov}(\mathbf{x}, \mathbf{y})$ is the covariance of the two images, and c_1 and c_2 are two parameters to stabilize the division with small denominators, the defaults are $c_1 = 0.0001, c_2 = 0.0009$.

In practice, SSIM is calculated on local windows rather than over the whole image. As in [36], we use a normalized 11×11 circular-symmetric Gaussian weighting function $\mathbf{w} = \{w_i \mid i = 1, 2, \cdots, N = 121\}$, with standard deviation of 1.5. As the result, $\mu_\mathbf{x}$, $\sigma_\mathbf{x}$ and $\text{cov}(\mathbf{x}, \mathbf{y})$ in the SSIM measure (4.1) are modified as

$$\mu_\mathbf{x} = \sum_{i=1}^{N} w_i x_i, \ \sigma_\mathbf{x} = \left(\sum_{i=1}^{N} w_i (x_i - \mu_\mathbf{x})^2 \right), \ \text{cov}(\mathbf{x}, \mathbf{y}) = \sum_{i=1}^{N} w_i (x_i - \mu_\mathbf{x})(y_i - \mu_\mathbf{y}).$$

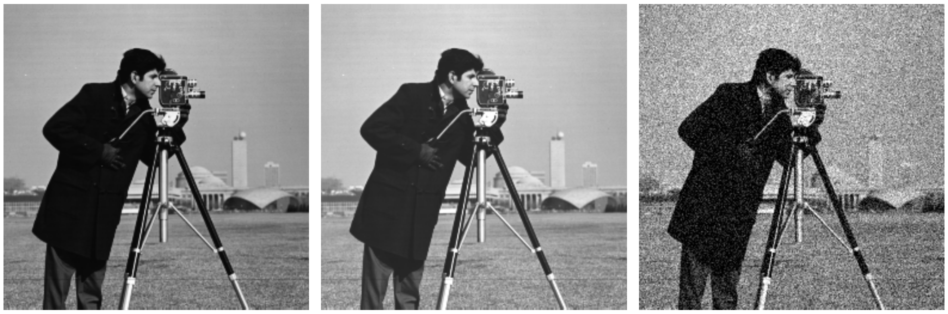
In order to get a single overall similarity measure of the two images, the MSSIM is computed by choosing the local window pixel-by-pixel,

$$\text{MSSIM}(I_t, I_{true}) = \frac{1}{mn} \sum_{i=1}^{mn} \text{SSIM}(\mathbf{x}_i, \mathbf{y}_i), \tag{4.2}$$

where $\mathbf{x}_i$ and $\mathbf{y}_i$ are the ith local windows, and $m \times n$ is the size of the true image, I_{true}, and restored image, I_t.

We also consider the traditional image-quality measurements, the peak signal-to-noise ratio (PSNR) [37],

$$\text{PSNR} = 20 \times \log_{10} \left(\frac{\max(I_\text{clean})}{\sqrt{\text{MSE}}} \right),$$



(a) reference image.

(b) Luminance shift image, $\mathrm{PSNR} = 19.17, \mathrm{MSSIM} = 0.91$

(c) White noise, $\mathrm{PSNR} = 19.5, \mathrm{MSSIM} = 0.31$.

Fig. 4.1: Comparison of MSSIM and PSNR. While the luminance-shift image (middle) and the one with white noise (right) have the same PSNR index, 19.5, the MSSIM of the luminance-shift image is 0.91, which is much larger than that for the image with white noise, 0.31.

where $\max(I_{\text{clean}})$ is the maximum possible value of clean image, $I_{\text{clean}} = 1$ in our experiments, and MSE is the mean squared error between the clean and noisy images. However, our experiments show that MSSIM is more suitable for measuring the quality of denoised images, see Figure 4.1. In this figure, both the luminance-shift image and the one with white noise have the same PSNR index, 19.17; however, by visual quality, the luminance-shift image is noise-free, and is much better than the one with white noise. This is shown from the MSSIM index: the MSSIM of the luminance-shift image is 0.91, while the MSSIM for the image with white noise is 0.31.

4.2. Experiments. We consider 5 common test images, also used in [13], and add various levels of Gaussian noise to these images to test the algorithms. We define the "noise level" of a test image as

$$\text{noise level} = \frac{||\text{noise}||_F}{||\text{clean image}||_F},$$

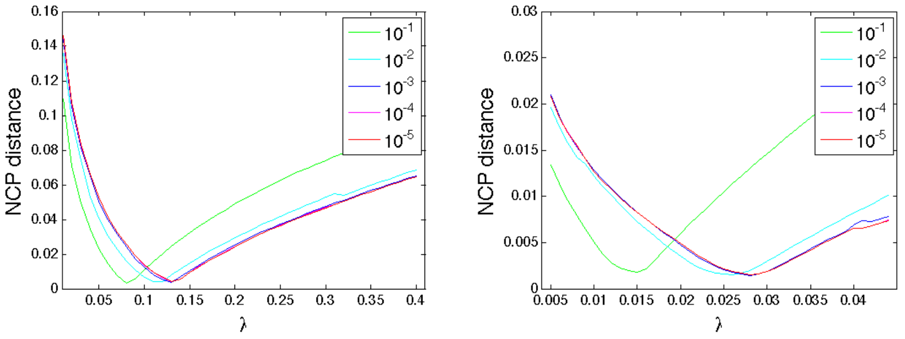
where $||\cdot||_F$ represents the Frobenius norm of $m \times n$ image. We include both the MSSIM and PSNR as measures of the restored image quality for the convenience of comparison with other papers, but note the results in Figure 4.1.

As mentioned previously, the quality of the restored image depends on the value of the regularization parameter. Thus, no matter which regularization scheme is employed, there will be an outer loop over the regularization parameter values. We initially consider two possibilities – choosing the λ that minimizes the CC functional vs. choosing the λ that minimizes the NCP functional. We show experimentally in the next section that each of these has a well-defined minimum, and use Brent's method to solve for that minimum (i.e. Brent-CC and Brent-NCP). For AOS applied to solve (2.3), the regularization parameter is the stopping time T, and the CC or the NCP approach can be used to determine this value. If solving either (2.5) (with AOS) or (2.6) (fixed point with BoxMG or AMG) is preferred to this approach, the regularization parameter that needs to be selected is λ.

The numerical experiments are outlined as follows

- In subsection 4.2.1, we fix the regularization approach as that of solving (2.6) using fixed point iteration with BoxMG as the linear solver. Then we investigate the use of the CC functional vs. the NCP functional to choose λ.
- In subsection 4.2.2, we fix the selection approach (i.e. outer iteration) as Brent-NCP and compare results given using three algorithms: AOS to solve (2.5) for each λ (we refer to this as AOS-R); solving (2.6) with fixed point iteration and BoxMG as the linear solver; solving (2.6) with fixed point iteration and AMG as the linear solver.
- To illustrate the applicability of our Brent-NCP regularization parameter selection approach, in subsection 4.2.3, we fix the selection approach as Brent-NCP and apply it to finding the regularization parameter (stopping time, T) for AOS applied to (2.3) and to finding the regularization parameter λ in Total Variation, a well-known denoising scheme. We compare these results with the traditional AOS scheme as presented in [23] and against the BoxMG results obtained in the previous subsection.
- We conclude with a comparison of the diffusion-based denoising techniques to the block-based denoising technique known as BM3D in Subsection 4.2.4.

For the anisotropic diffusion approach based on (2.6), two important technical parameters need to be fixed: the outer stopping tolerance for the fixed-point iteration, $\mathrm{tol}_{\mathrm{f}p}$, and the inner stopping tolerance for the multigrid solvers for each linearization, $\mathrm{tol}_{\mathrm{m}g}$. In Figure 4.2, we investigate the effects of varying $\mathrm{tol}_{\mathrm{f}p}$ with $\mathrm{tol}_{\mathrm{m}g}$ fixed at 0.1. Note that while the optimal λ changes significantly with $\mathrm{tol}_{\mathrm{f}p}$, the quality of the restored image, as measured by the NCP distance is much less sensitive. Similarly, Table 4.1 compares the effects of the inner stopping tolerance, $\mathrm{tol}_{\mathrm{m}g}$, with $\mathrm{tol}_{\mathrm{f}p}$ fixed at 10^{-3}. Note that the computed regularization parameter is essentially unchanged by choosing larger values of $\mathrm{tol}_{\mathrm{m}g}$. Thus, we take $\mathrm{tol}_{\mathrm{m}g} = 0.1$ in the experiments here to improve the overall efficiency of the approach, in combination with $\mathrm{tol}_{\mathrm{f}p} = 10^{-3}$.



(a) BoxMG, noise level = 0.3

(b) BoxMG, noise level = 1.0

Fig. 4.2: Test for tolerance of fixed-point iteration, $\mathrm{tol}_{\mathrm{fp}}$, NCP-$\lambda$ plots for cameraman image. For given regularization parameter, λ, on the x-axis, the y-axis shows the NCP distance measurements of the restored images computed using the corresponding λ. As $\mathrm{tol}_{\mathrm{fp}}$ decreases from 10^{-1} to 10^{-5}, the NCP-λ plot converges to a limiting curve for different noise levels, 0.3 (left) and 1.0 (right).

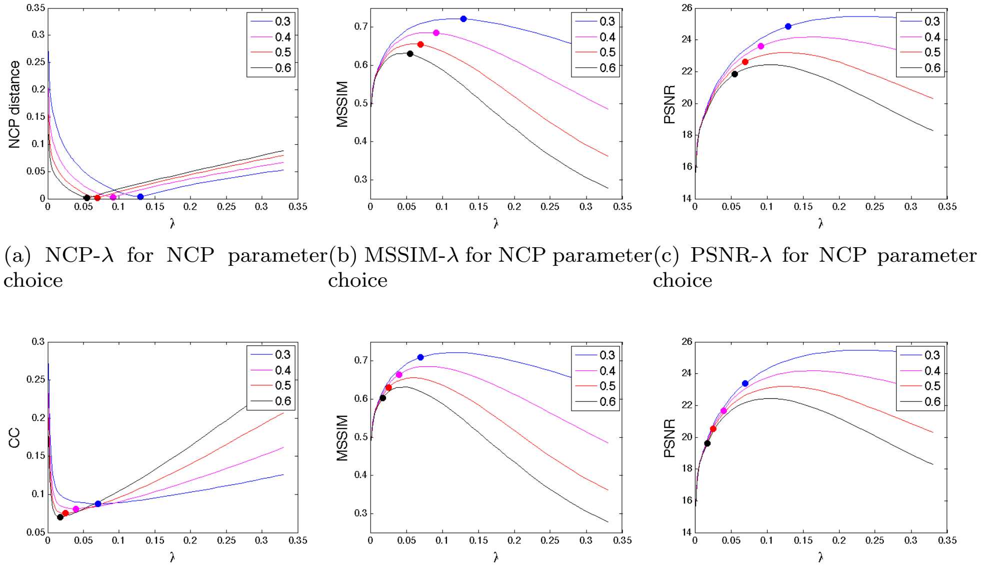
Table 4.1: Comparison of the computed regularization parameters and the computational work in the Brent-NCP algorithm using BoxMG to solve the linearized systems.

noise level	tol_{mg}	Brent	linearization	V-cycles	time(s)	λ_{opt}	relative
		steps	steps				change(%)
0.3	10^{-6}	16	153	1351	21.5	1.30e-01	0
	10^{-5}	16	153	1052	16.2	1.30e-01	2.51e-06
	10^{-4}	16	153	787	13.6	1.30e-01	4.89e-04
	10^{-3}	16	154	510	11.1	1.30e-01	2.49e-02
	10^{-2}	16	154	322	9.2	1.30e-01	2.72e-03
	10^{-1}	15	146	158	7.3	1.30e-01	2.38e-01
1.0	10^{-6}	16	213	1926	26.7	2.80e-02	0
	10^{-5}	16	213	1511	22.7	2.80e-02	1.81e-05
	10^{-4}	16	213	1098	18.6	2.80e-02	2.12e-04
	10^{-3}	16	213	737	15.2	2.80e-02	3.49e-03
	10^{-2}	16	215	451	12.6	2.80e-02	4.51e-01
	10^{-1}	16	221	233	10.8	2.80e-02	4.95e-01

4.2.1. Optimizing CC vs NCP for choosing λ. We first show experimentally that there are unique minimizers for both (3.5) and (3.3), and that employing Brent's method achieves these minimizers. We solve our objective function (2.6) using the fixed-point iteration with BoxMG as the inner linear solver, where the regularization parameters are computed using Brent's method. For the CC method, instead of computing $\mathcal{N}(\lambda)$ in Steps 4 and 12, we calculate the correlation coefficient as in Equation (3.3). Figures 4.3 and 4.4 show the results for two test images: cameraman and fingerprint, with noise level varying from 0.3 to 0.6. The dots shown in Figures 4.3 and 4.4 are the results computed using Brent's method. It is clear that the computed solutions are the minimizers of NCP distance and correlation coefficient functions.

Figures 4.3 and 4.4 also show the MSSIM and PSNR measurements of the restored images. Compared to the CC method, one of the advantages of the NCP approach is that the MSSIM and PSNR measurements of the restored images corresponding to the minimizers of the NCP distance are close to the maximum values of MSSIM and PSNR that are achieved. While this is also nearly true for the CC method applied to the cameraman image, it is obviously not the case for the fingerprint image, which has more texture information than the cameraman image. This experiment shows that the NCP method is more stable than the CC method and that minimizing the NCP correlates nicely and consistently across images with achieving large MSSIM or PSNR measures of the restored images.

Next, we compare the restored image quality computed using these regularization parameter-choice methods. The results are shown in Table 4.2. For the images with little texture information, such as the cameraman image, the results using CC are comparable to those using NCP, especially for low noise levels. However, for large noise levels, or images with lots of texture information, NCP clearly outperforms CC. This is especially noticeable in the fingerprint image; with a noise level of 1.0, the regularization parameter chosen based on the NCP criterion yields a significant improvement on that chosen based on the CC criterion, particularly when considering their MSSIM measures of 0.40 and 0.11.



(a) NCP-λ for NCP parameter choice (b) MSSIM-λ for NCP parameter choice (c) PSNR-λ for NCP parameter choice

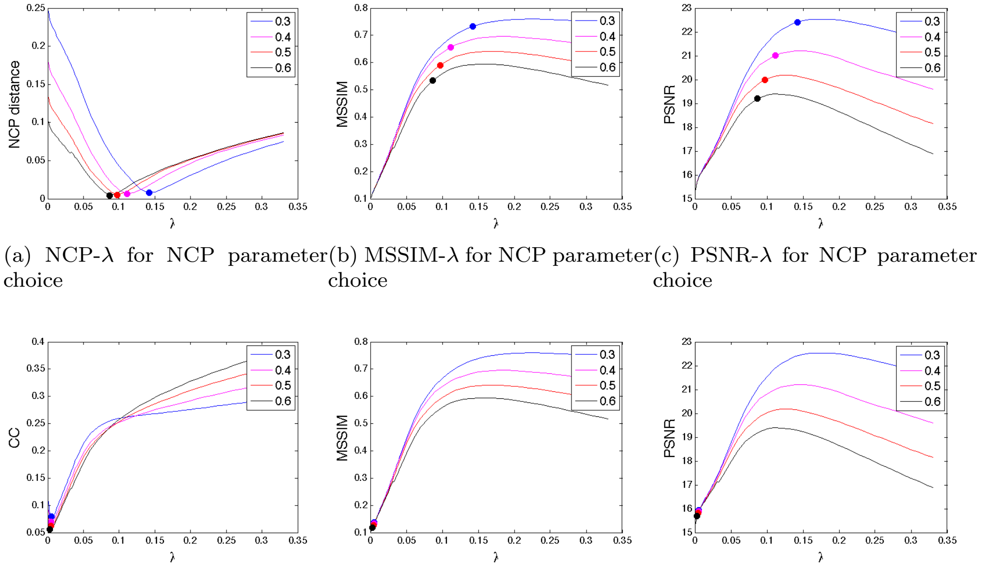
(d) CC-λ for CC parameter choice (e) MSSIM-λ for CC parameter choice (f) PSNR-λ for CC parameter choice

Fig. 4.3: Results for cameraman image, where the noise level varies from 0.3 to 0.6. The dots are the restored image measurements computed using the optimal regularization parameters, λ, which are computed using Brent-NCP (top) and Brent-CC (bottom).

Table 4.3: Comparison of computational time and results in Brent-NCP algorithm using two different multigrid solvers: AMG and BoxMG for the cameraman image. This table shows that the number of Brent's method and linearization steps are nearly the same for both solvers, as are the computed regularization parameters. However, the computational time for BoxMG is only about 1/6 of that of AMG due the expensive setup phase and indirect addressing used within AMG.

noise level	optimal λ		time		brent steps		linearization step		total V-cycles	
	AMG	BoxMG	AMG	BoxMG	AMG	BoxMG	AMG	BoxMG	AMG	BoxMG
0.1	4.61e-1	4.61e-1	17.3	2.7	10	10	53	53	53	94
0.2	2.04e-1	2.05e-1	25.0	4.9	11	12	87	96	107	198
0.3	1.30e-1	1.30e-1	44.0	7.4	16	15	153	146	246	322
0.4	9.16e-2	9.16e-2	42.2	7.5	14	14	151	151	250	314
0.5	6.98e-2	6.98e-2	40.8	7.2	13	13	144	144	233	299
0.6	5.49e-2	5.51e-2	41.2	7.0	12	12	142	141	229	290
0.7	4.47e-2	4.48e-2	51.6	8.7	14	14	179	177	296	401
0.8	3.74e-2	3.75e-2	60.2	9.9	15	15	205	200	355	389
0.9	3.19e-2	3.20e-2	64.4	10.7	15	15	218	214	378	435
1.0	2.79e-2	2.80e-2	65.8	10.8	15	16	220	221	392	451

4.2.2. Comparison of AMG, BoxMG, and AOS for regularized diffusion. We first compare the restored images computed using two different multigrid solvers, BoxMG and AMG, using both solvers in the computation of the optimal reg-



(a) NCP-λ for NCP parameter choice
(b) MSSIM-λ for NCP parameter choice
(c) PSNR-λ for NCP parameter choice

(d) CC-λ for CC parameter choice
(e) MSSIM-λ for CC parameter choice
(f) PSNR-λ for CC parameter choice

Fig. 4.4: Results for fingerprint image, where the noise level varies from 0.3 to 0.6. The dots are the restored image measurements computed using the optimal regularization parameters, λ, which are computed using Brent-NCP (top) and Brent-CC (bottom).

ularization parameters using Brent's method for the cameraman image with different noise levels. The results are shown in Table 4.3, where we see that the number of Brent's method and linearization steps are nearly the same for both solvers. This is also true for the computed regularization parameters. Furthermore, from the results in Table 4.4, we can see that the MSSIM and PSNR measurements of restored images when using BoxMG and AMG to solve the anisotropic diffusion equation (2.6) are almost the same. Note that while the iteration counts for AMG look better than those for BoxMG, BoxMG is computationally faster. As shown in Table 4.3, BoxMG is about 6 times faster than AMG in terms of computational time, due to the combination of the more expensive setup phase within AMG and its use of unstructured storage and indirect addressing. While AMG is known to be very robust and effective for highly discontinuous diffusivities, these results show that the added costs required for AMG do not pay off in this situation.

As mentioned in Section 2, the AOS scheme can be applied directly to solve the time-dependent regularized anisotropic diffusion equation (2.5). We call this scheme AOS-R. We use an outer iteration to find λ by the Brent-NCP Algorithm 2. In each iteration with fixed λ, we solve for the steady-state solution of the regularized diffusion equation with AOS-R, with $\tau = 0.5$. This inner iteration stops when the relative difference between two consecutive steps measured in the Frobenius norm becomes less than 10^{-5}. Numerical experiments indicate that smaller timesteps or a more accurate stopping criterion do not improve these results. The quality measures of the results are shown in Table 4.4. From this table, we can see that both the AMG and BoxMG solvers for (2.6) (reported results are for λ values chosen using Brent-NCP as the outer wrapper in each case as well) outperform the AOS-R for (2.5) in terms of accuracy. The results suggest that solving somewhat inaccurately over a possibly large number of time steps permits some unintentional smoothing into the process of solving for the steady state solution, whereas solving (2.6) by fewer, and more accurate, fixed point steps, prevents this problem (see also [5]). Another interesting feature to note is that the PSNR values for the AOS-R results are very static with respect to noise level for each test problem. We consider this further evidence of the pitfalls of using PSNR to measure restored image quality (see also examples in [37]).

4.2.3. Comparison with unregularized AOS and TV. We test the applicability of the Brent-NCP Algorithm 2 by computing the diffusion stopping time in (2.3) for the AOS scheme and the regularization parameter, λ, for TV denoising.

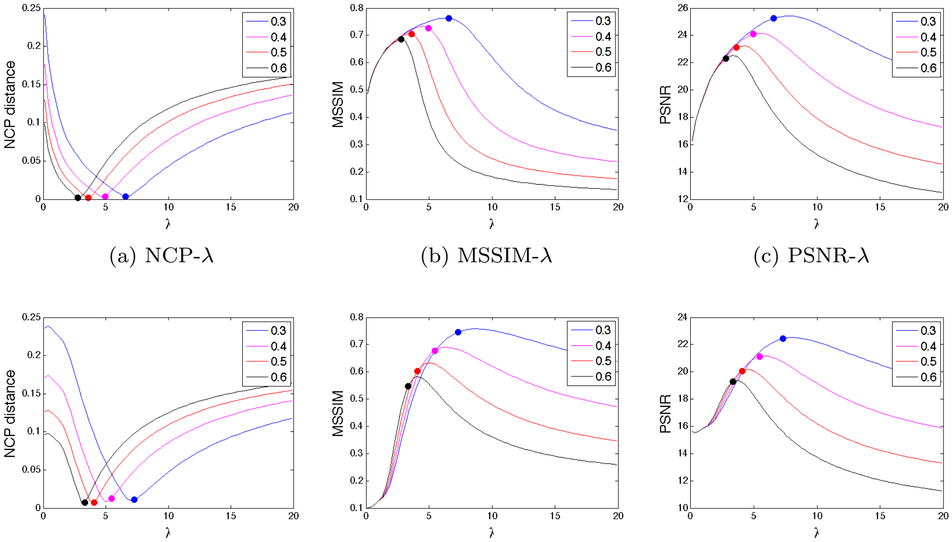
For the AOS scheme, in the experiments of [23, 42], the authors use fixed time steps, τ, to compute the restored images. If we use a fixed time step, say $\tau = 0.5$, the computational time for the high noise level images will become very large compared with the other methods. Moreover, in [42], Weickert et al. point out that AOS with semi-implicit time-stepping is stable for all time steps. Therefore, instead of fixing the time step, τ, we fix the number of time steps to be 10, i.e. the time step, $\tau = \frac{T}{10}$, where T is the stopping time returned by the Brent's method. The reason we choose $\tau = \frac{T}{10}$ is illustrated later in the discussion.

The objective function for TV is

$$\min_{I} \mathrm{TV}(I) + \frac{\lambda}{2}\|I - I_0\|_2^2, \tag{4.3}$$

where $\mathrm{TV}(I) = \int_\Omega |\nabla I|^2$, and λ is the regularization parameter [30]. Here, we use the algorithm proposed by Chambolle [11] to solve (4.3). Different regularization parameters, λ, have tremendous impact on the restored image quality. Traditionally, the regularization parameter is chosen by trial and error. Recent research into better ways to choose the regularization parameter has included methods based on image geometry or local variance estimators [15, 33]. Figure 4.5 shows the MSSIM and PSNR measurement of the restored images computed using TV denoising, where the regularization parameters are computed using the Brent-NCP method replacing the anisotropic diffusion solves in Step 4 and 12 with the TV minimization in (4.3). Note that minimizing the NCP correlates nicely with achieving large MSSIM or PSNR measures of the restored images.

Table 4.5 gives comparisons of the unregularized AOS scheme for (2.3) (AOS-U) with diffusion time, T, chosen by the Brent-NCP algorithm, TV denoising with regularization parameter, λ, chosen by the Brent-NCP algorithm, and the traditional AOS scheme (AOS-T) where the diffusion time, T, is chosen as in [23], with $\tau = 0.5s$. For the AOS scheme with the Brent-NCP method for choosing the stopping time in (2.3), we found that 10 discrete time steps were sufficient to give restored images of quality comparable to the other two methods in our results. The quality of the restored images using anisotropic diffusion with the proposed algorithm is comparable to that of the images computing using the AOS-U and TV denoising. As shown in Figure 4.6, it is difficult to see any difference in the detail regions shown for the restored Barbara images. They all are clearly better than the results by AOS-T, especially in the MSSIM measures. One possible way to enhance the restored image quality using the AOS-T scheme is to use a very small time step, τ. However, due to the added cost of the additional time steps, the execution time will become much longer than the other schemes. Since all these methods are greatly dependent on the choice of regularization parameters, which are all computed using the Brent-NCP algorithm, this table also shows the broad applicability of the Brent-NCP algorithm in choosing regularization parameters.



(a) NCP-λ

(b) MSSIM-λ

(c) PSNR-λ

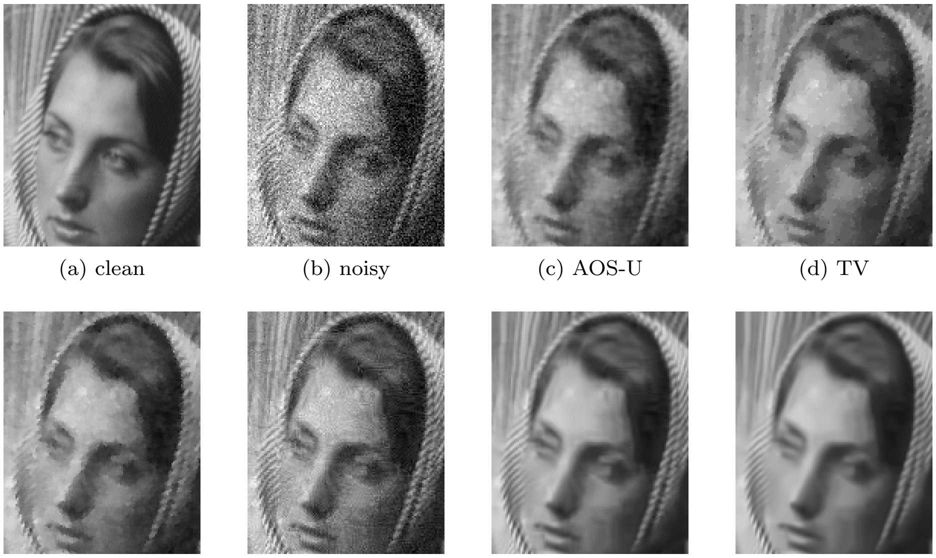
(d) NCP-λ

(e) MSSIM-λ

(f) PSNR-λ

Fig. 4.5: Results for cameraman (top) and fingerprint (bottom) images, where the noise level varies from 0.3 to 0.6. The dots are the restored image measurements computed using the optimal regularization parameters, λ, which are computed using Brent-NCP and TV denoising.

4.2.4. Comparison with block-based denoising. While diffusion-based denoising techniques are common in the literature, many other approaches are also possible. Among them, BM3D proves to be very effective and is one of the best denoising methods in the literature [13]. BM3D is a block-based approach that collects the local information in a noisy image and groups similar 2D image fragments together into 3D data arrays. Then, a collaborative filtering technique is used to deal with these 3D groups. Table 4.6 compares BM3D with the anisotropic diffusion approach presented here. One of the disadvantages of the BM3D algorithm is that it requires the user to input the estimated noise level. Given an accurate estimated noise level, the restored images from BM3D have better quality than those generated by anisotropic diffusion in terms of both MSSIM and PSNR. However, noise estimation itself is a difficult research area [22]. If the input noise level is not accurate, the restored images can have a bad quality, especially if the noise level is underestimated, see Figure 4.6. This is also shown in Table 4.6. We use different noise level inputs: the exact noise level, n_0, an underestimated noise level, $0.7 \times n_0$, and an overestimated noise level, $1.3 \times n_0$. The MSSIM and PSNR measurements decrease substantially for the underestimated case. Here, we point out that, in practice, it is very difficult to accurately estimate the true noise level. For this reason, it is not fair to compare between anisotropic diffusion denoising and BM3D by simply looking at the measurements without considering the algorithms requirements.



(a) clean

(b) noisy

(c) AOS-U

(d) TV

(e) AD

(f) BM3D with under-estimated noise level

(g) BM3D with exact noise level

(h) BM3D with over-estimated noise level

Fig. 4.6: Detail of restored Barbara image by different methods. The noise level in the noisy image (b) is 0.2. The regularization parameters for AOS-U, TV, and AD are computed use Brent-NCP method. The MSSIM and PSNR measurements for the restored images are shown in Tables 4.4, 4.5 and 4.6.

5. Conclusion. The goals of this paper are to investigate the application of the multigrid algorithm to image restoration and to introduce a new regularization parameter-choice approach. After reviewing the anisotropic diffusion equation methodology and its application to image restoration, we solve a regularized anisotropic diffusion equation (2.6), which is not only well-posed, but also has a nontrivial steady-state solution, using a fixed-point multigrid iteration. In order to make the algorithm more adaptive and efficient, we introduce a new automatic regularization parameter-choice method. It combines Brent's method and the NCP information of image misfits. The application of the Brent-NCP parameter-choice Algorithm 2 is not limited to anisotropic-diffusion based denoising. It is shown to have applicability to a very broad range of parameter-dependent denoising algorithms.

REFERENCES

[1] S. Acton. Multigrid anisotropic diffusion. *IEEE Transactions on Image Processing*, 7(3):280–290, 1998.

[2] R. Alcouffe, A. Brandt, J. Dendy Jr., and J. Painter. The multi-grid method for the diffusion equation with strongly discontinuous coefficients. *SIAM Journal on Scientific and Statistical Computing*, 2(4):430–454, 1981.

[3] K. Atkinson. *An Introduction to Numerical Analysis.* Wiley, 2 edition, 1989.

[4] G. Aubert and P. Kornprobst. *Mathematical Problems in Image Processing: Partial Differential Equations and the Calculus of Variations (second edition)*, volume 147 of *Applied Mathematical Sciences*. Springer-Verlag, 2006.

[5] D. Barash, M. Israeli, and R. Kimmel. An accurate operator splitting scheme for nonlinear diffusion filtering. In *Scale-Space Theories in Computer Vision*, pages 281–289, 2001.

[6] A. Brandt, S. McCormick, and J. Ruge. Algebraic multigrid (AMG) for sparse matrix equations. In D. J. Evans, editor, *Sparsity and its Applications*, pages 257–284. CAMBRIDGE, 1984.

[7] W. Briggs, V. Henson, and S. McCormick. *A multigrid tutorial (2nd ed.)*. Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 2000.

[8] A. Buades, B. Coll, and J.M. Morel. On image denoising methods. Technical report, Technical Note, CMLA (Centre de Mathematiques et de Leurs Applications), 2004.

[9] J. Canny. A computational approach to edge detection. *IEEE Transactions on Pattern Anal. Mach. Intell.*, 8(6):679–698, 1986.

[10] F. Catté, P.L. Lions, J.M. Morel, and T. Coll. Image selective smoothing and edge detection by nonlinear diffusion. *SIAM J. Numer. Anal.*, 29(1):182–193, 1992.

[11] A. Chambolle. An algorithm for total variation minimization and applications. *Journal of Mathematical Imaging and Vision*, 20(1-2):89–97, 2004.

[12] K. Chen and J. Savage. An accelerated algebraic multigrid algorithm for total-variation denoising. *Bit Numerical Mathematics*, 47(2):277–296, 6 2007.

[13] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian. Image denoising with block-matching and 3D filtering. In *Proc. SPIE Electronic Imaging: Algorithms and Systems V*, volume 6064, 2006.

[14] J. Dendy. Black box multigrid. *Journal of Computational Physics*, 48(3):366 – 386, 1982.

[15] Y. Dong and M. Hintermüller. Multi-scale total variation with automated regularization parameter selection for color image restoration. In *Scale Space and Variational Methods in Computer Vision*, volume 5567 of *Lecture Notes in Computer Science*, pages 271–281. Springer Berlin / Heidelberg, 2009.

[16] J.M. Duarte-Carvajalino, G. Sapiro, M. Velez-Reyes, and P.E. Castillo. Multiscale representation and segmentation of hyperspectral imagery using geometric partial differential equations and algebraic multigrid methods. *IEEE Transactions on Geoscience and Remote Sensing*, 46(8):2418–2434, Aug. 2008.

[17] W. Fuller. *Introduction to Statistical Time Series, Second Edition*. John Wiley, New York, NY, 1996.

[18] G. Gilboa. Nonlinear scale space with spatially varying stopping time. *IEEE Trans. Pattern Anal. Mach. Intell.*, 30:2175–2187, December 2008.

[19] P.C. Hansen, M. Kilmer, and R. Kjeldsen. Exploiting residual information in the parameter choice for discrete ill-posed problems. *BIT*, 46:41–59, 2006.

[20] S. Kichenassamy. The Perona-Malik paradox. *SIAM J. Appl. Math.*, 57(5):1328–1342, 1997.

[21] R. Kimmel and I. Yavneh. An algebraic multigrid approach for image analysis. *SIAM J. Sci. Comput*, 24:1218–1231, 2003.

[22] C. Liu, W. Freeman, R. Szeliski, and S. Kang. Noise estimation from a single image. *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Volume 1 CVPR06*, 1:901–908, 2006.

[23] P. Mrázek. Selection of optimal stopping time for nonlinear diffusion filtering. In *Proceedings of the Third International Conference on Scale-Space and Morphology in Computer Vision*, Scale-Space 01, pages 290–298. Springer-Verlag, 2001.

[24] K. Nordström. Biased anisotropic diffusion–a unified regularization and diffusion approach to edge detection. Technical Report UCB/CSD-89-514, EECS Department, University of California, Berkeley, 5 1989.

[25] G. Papandreou and P. Maragos. A cross-validatory statistical approach to scale selection for image denoising by nonlinear diffusion. In *Computer Vision and Pattern Recognition*, pages 625–630, 2005.

[26] G. Papandreou and P. Maragos. Multigrid geometric active contour models. *Image Processing, IEEE Transactions on*, 16(1):229 –240, jan. 2007.

[27] P. Perona and J. Malik. Scale-space and edge detection using anisotropic diffusion. *Proceedings of IEEE Computer Society workshop on Computer Vision*, pages 16–22, 1987.

[28] P. Perona and J. Malik. Scale-space and edge detection using anisotropic diffusion. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(7):629–639, Jul 1990.

[29] W. Press, B. Flannery, S. Teukolsky, and W. Vetterling. *Numerical Recipes in C: The Art of Scientific Computing*. Cambridge University Press, 2 edition, October 1992.

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

[31] J. Ruge and K. Stüben. Algebraic multigrid. In S. F. Mccormick, editor, *Multigrid Methods*, pages 73–130, Philidelphia, Pennsylvania, 1987. SIAM.

[32] B. Rust and D. O'Leary. Residual periodograms for choosing regularization parameters for ill-posed problems. *Inverse Problems*, 24(3):034005 (30pp), 2008.

[33] D. Strong, J.-F. Aujol, and T. Chan. Scale recognition, regularization parameter selection, and Meyer's G norm in total variation regularization. *Multiscale Modeling and Simulation*, 5(1):273–303, 2006.

[34] B. Tomasik, I. Melo, G. Torrieri, S. Vogel, and M. Bleicher. The use of Kolmogorov-Smirnov test in event-by-event analysis. *Nucl. Phys.*, A830:195c–198c, 2009.

[35] U. Trottenberg, C. Oosterlee, and A. Schüller. *Multigrid*. Academic Press, San Diego, U.S.A., 2001.

[36] Z. Wang, A. Bovik, H. Sheikh, and E. Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4):600–612, April 2004.

[37] Z. Wang and A.C. Bovik. Mean squared error: Love it or leave it? a new look at signal fidelity measures. *IEEE Signal Processing Magazine*, 26(1):98–117, January 2009.

[38] J. Weickert. *Anisotropic Diffusion in Image Processing*. PhD thesis, Universität Kaiserslautern, Germany, 1996.

[39] J. Weickert. A review of nonlinear diffusion filtering. In *SCALE-SPACE '97: Proceedings of the First International Conference on Scale-Space Theory in Computer Vision*, pages 3–28, London, UK, 1997. Springer-Verlag.

[40] J. Weickert. Coherence-enhancing diffusion of colour images. *Image and Vision Computing*, 17(3-4):201 – 212, 1999.

[41] J. Weickert and B. Benhamouda. A semidiscrete nonlinear scale-space theory and its relation to the Perona-Malik paradox. In *F. Solina (Ed.), Advances in computer vision*, pages 1–10. Springer, 1997.

[42] J. Weickert, B.M. Ter Haar Romeny, and M.A. Viergever. Efficient and reliable schemes for nonlinear diffusion filtering. *IEEE Transactions on Image Processing*, 7:398–410, 1998.

[43] J. Weickert and H. Scharr. A scheme for coherence-enhancing diffusion filtering with optimized rotation invariance. *Journal of Visual Communication and Image Representation*, 13(1-2):103 – 118, 2002.

[44] K. Yen and R. Johnston. The ineffectiveness of the correlation coefficient for image comparisons. Technical report, Los Alamos National Labs, 1996.

Table 4.2: Comparison of NCP and CC criteria for choosing the regularization parameter, λ. All the results are computed using Brent's algorithm and fixed-point iteration with the BoxMG solver.

images	noise level	MSSIM			PSNR		
		noisy	CC	NCP	noisy	CC	NCP
cameraman	0.1	0.52	0.84	0.84	25.5	30.2	29.8
	0.2	0.31	0.76	0.77	19.5	25.7	26.7
	0.3	0.21	0.71	0.72	16.0	23.4	24.9
	0.4	0.16	0.66	0.68	13.5	21.7	23.6
	0.5	0.12	0.63	0.65	11.5	20.5	22.6
	0.6	0.10	0.60	0.63	10.0	19.6	21.9
	0.7	0.08	0.59	0.61	8.6	19.3	21.2
	0.8	0.06	0.58	0.59	7.5	19.1	20.7
	0.9	0.05	0.58	0.58	6.4	18.9	20.3
	1.0	0.04	0.57	0.57	5.5	18.8	20.0
house	0.1	0.44	0.84	0.84	24.3	31.8	31.5
	0.2	0.23	0.78	0.78	18.3	28.2	28.2
	0.3	0.14	0.73	0.74	14.8	25.4	26.3
	0.4	0.10	0.68	0.71	12.3	22.7	25.0
	0.5	0.07	0.64	0.69	10.3	20.9	24.1
	0.6	0.05	0.64	0.67	8.8	20.5	23.3
	0.7	0.04	0.63	0.65	7.4	19.8	22.6
	0.8	0.03	0.62	0.64	6.3	19.7	22.1
	0.9	0.03	0.62	0.63	5.2	19.5	21.6
	1.0	0.02	0.62	0.63	4.3	19.4	21.1
barbara	0.1	0.62	0.72	0.84	25.6	24.5	28.7
	0.2	0.38	0.66	0.71	19.6	23.5	24.9
	0.3	0.26	0.62	0.64	16.0	22.9	23.5
	0.4	0.19	0.59	0.60	13.5	22.4	22.8
	0.5	0.14	0.57	0.58	11.6	21.9	22.3
	0.6	0.10	0.55	0.56	10.0	21.4	21.8
	0.7	0.08	0.53	0.54	8.7	20.9	21.5
	0.8	0.07	0.51	0.52	7.5	20.5	21.2
	0.9	0.05	0.50	0.51	6.5	20.1	20.9
	1.0	0.04	0.49	0.50	5.6	19.6	20.6
boat	0.1	0.57	0.80	0.82	25.3	30.0	30.4
	0.2	0.31	0.67	0.72	19.3	25.9	27.4
	0.3	0.19	0.60	0.66	15.8	23.9	25.8
	0.4	0.13	0.56	0.62	13.3	22.6	24.7
	0.5	0.10	0.53	0.59	11.4	21.9	23.9
	0.6	0.07	0.52	0.57	9.8	21.5	23.3
	0.7	0.06	0.47	0.55	8.4	19.5	22.8
	0.8	0.04	0.47	0.53	7.3	19.3	22.3
	0.9	0.04	0.46	0.52	6.3	19.1	21.9
	1.0	0.03	0.46	0.51	5.3	18.8	21.6
fingerprint	0.1	0.84	0.91	0.92	24.6	27.5	28.2
	0.2	0.61	0.83	0.82	18.6	24.3	24.5
	0.3	0.43	0.14	0.73	15.0	15.9	22.4
	0.4	0.31	0.13	0.66	12.5	15.9	21.0
	0.5	0.23	0.13	0.59	10.6	15.8	20.0
	0.6	0.18	0.12	0.53	9.0	15.7	19.2
	0.7	0.14	0.12	0.49	7.7	15.7	18.6
	0.8	0.11	0.12	0.46	6.5	15.6	18.2
	0.9	0.09	0.11	0.43	5.5	15.6	17.9
	1.0	0.08	0.11	0.40	4.6	15.6	17.6

Table 4.4: Comparison of restored images using different methods: BoxMG and AMG solvers for (2.6), and regularized AOS, AOS-R. For the AOS-R scheme, we take time step, $\tau = 0.5$, and use a relative difference stopping criterion of $relerr = 10^{-5}$.

images	noise level	MSSIM				PSNR			
		noisy	BoxMG	AMG	AOS-R	noisy	BoxMG	AMG	AOS-R
cameraman	0.1	0.52	0.84	0.84	0.55	25.5	29.8	29.8	11.2
	0.2	0.31	0.77	0.77	0.56	19.5	26.7	26.7	11.3
	0.3	0.21	0.72	0.72	0.53	16.0	24.9	24.9	11.3
	0.4	0.16	0.68	0.68	0.48	13.5	23.6	23.6	11.3
	0.5	0.12	0.65	0.65	0.41	11.5	22.6	22.6	11.3
	0.6	0.10	0.63	0.63	0.33	10.0	21.9	21.9	11.3
	0.7	0.08	0.61	0.61	0.26	8.6	21.2	21.2	11.2
	0.8	0.06	0.59	0.59	0.22	7.5	20.7	20.7	11.1
	0.9	0.05	0.58	0.58	0.20	6.4	20.3	20.3	11.1
	1.0	0.04	0.57	0.57	0.21	5.5	20.0	20.0	11.1
house	0.1	0.44	0.84	0.84	0.59	24.3	31.5	31.5	10.2
	0.2	0.23	0.78	0.78	0.58	18.3	28.2	28.2	10.3
	0.3	0.14	0.74	0.74	0.52	14.8	26.3	26.3	10.3
	0.4	0.10	0.71	0.71	0.41	12.3	25.0	25.0	10.3
	0.5	0.07	0.69	0.69	0.32	10.3	24.1	24.1	10.2
	0.6	0.05	0.67	0.67	0.30	8.8	23.3	23.3	10.2
	0.7	0.04	0.65	0.65	0.34	7.4	22.6	22.6	10.2
	0.8	0.03	0.64	0.64	0.37	6.3	22.1	22.1	10.2
	0.9	0.03	0.63	0.63	0.38	5.2	21.6	21.6	10.2
	1.0	0.02	0.63	0.63	0.40	4.3	21.1	21.1	10.2
barbara	0.1	0.62	0.84	0.84	0.48	25.6	28.7	28.7	11.4
	0.2	0.38	0.71	0.71	0.48	19.6	24.9	24.9	11.4
	0.3	0.26	0.64	0.64	0.46	16.0	23.5	23.5	11.4
	0.4	0.19	0.60	0.60	0.43	13.5	22.8	22.8	11.4
	0.5	0.14	0.58	0.58	0.37	11.6	22.3	22.3	11.4
	0.6	0.10	0.56	0.56	0.31	10.0	21.8	21.8	11.3
	0.7	0.08	0.54	0.54	0.26	8.7	21.5	21.5	11.2
	0.8	0.07	0.52	0.52	0.22	7.5	21.2	21.2	11.1
	0.9	0.05	0.51	0.51	0.21	6.5	20.9	20.9	11.1
	1.0	0.04	0.50	0.50	0.23	5.6	20.6	20.6	11.2
boat	0.1	0.57	0.82	0.82	0.51	25.3	30.4	30.4	11.2
	0.2	0.31	0.72	0.72	0.51	19.3	27.4	27.4	11.2
	0.3	0.19	0.66	0.66	0.49	15.8	25.8	25.8	11.2
	0.4	0.13	0.62	0.62	0.43	13.3	24.7	24.7	11.2
	0.5	0.10	0.59	0.59	0.35	11.4	23.9	23.9	11.2
	0.6	0.07	0.57	0.57	0.29	9.8	23.3	23.3	11.1
	0.7	0.06	0.55	0.55	0.27	8.4	22.8	22.8	11.1
	0.8	0.04	0.53	0.53	0.29	7.3	22.3	22.3	11.1
	0.9	0.04	0.52	0.52	0.30	6.3	21.9	21.9	11.1
	1.0	0.03	0.51	0.51	0.31	5.3	21.6	21.6	11.1
fingerprint	0.1	0.84	0.92	0.92	0.51	24.6	28.2	28.2	10.4
	0.2	0.61	0.82	0.82	0.51	18.6	24.5	24.5	10.4
	0.3	0.43	0.73	0.73	0.50	15.0	22.4	22.4	10.4
	0.4	0.31	0.66	0.66	0.47	12.5	21.0	21.0	10.4
	0.5	0.23	0.59	0.59	0.44	10.6	20.0	20.0	10.3
	0.6	0.18	0.53	0.53	0.40	9.0	19.2	19.2	10.2
	0.7	0.14	0.49	0.49	0.38	7.7	18.6	18.6	10.2
	0.8	0.11	0.46	0.46	0.37	6.5	18.2	18.2	10.2
	0.9	0.09	0.43	0.43	0.35	5.5	17.9	17.9	10.2
	1.0	0.08	0.40	0.40	0.33	4.6	17.6	17.6	10.1

Table 4.5: Comparison of restored images using different methods: unregularized AOS, AOS-U, the traditional AOS scheme, AOS-T, where the stopping time is found by minimizing the CC measure, and TV as in (4.3).

images	noise level	MSSIM					PSNR				
		noisy	BoxMG	AOS-U	AOS-T	TV	noisy	BoxMG	AOS-U	AOS-T	TV
cameraman	0.1	0.52	0.84	0.82	0.73	0.88	25.5	29.8	28.2	29.0	30.5
	0.2	0.31	0.77	0.73	0.69	0.81	19.5	26.7	25.1	25.7	27.1
	0.3	0.21	0.72	0.67	0.62	0.76	16.0	24.9	23.8	24.1	25.2
	0.4	0.16	0.68	0.64	0.57	0.73	13.5	23.6	22.7	23.1	24.1
	0.5	0.12	0.65	0.61	0.53	0.71	11.5	22.6	22.0	22.3	23.1
	0.6	0.10	0.63	0.58	0.50	0.69	10.0	21.9	21.5	21.7	22.3
	0.7	0.08	0.61	0.58	0.47	0.67	8.6	21.2	20.9	21.2	21.7
	0.8	0.06	0.59	0.57	0.45	0.65	7.5	20.7	20.5	20.7	21.2
	0.9	0.05	0.58	0.56	0.44	0.64	6.4	20.3	20.2	20.3	20.9
	1.0	0.04	0.57	0.54	0.42	0.63	5.5	20.0	19.9	20.0	20.4
house	0.1	0.44	0.84	0.83	0.68	0.85	24.3	31.5	30.6	29.2	32.1
	0.2	0.23	0.78	0.76	0.67	0.80	18.3	28.2	27.6	27.7	29.0
	0.3	0.14	0.74	0.72	0.60	0.77	14.8	26.3	26.0	25.9	27.1
	0.4	0.10	0.71	0.69	0.60	0.75	12.3	25.0	24.9	25.1	25.8
	0.5	0.07	0.69	0.67	0.56	0.73	10.3	24.1	24.0	24.1	24.8
	0.6	0.05	0.67	0.66	0.53	0.71	8.8	23.3	23.4	23.3	23.8
	0.7	0.04	0.65	0.64	0.51	0.69	7.4	22.6	22.8	22.6	23.1
	0.8	0.03	0.64	0.63	0.51	0.68	6.3	22.1	22.2	22.3	22.5
	0.9	0.03	0.63	0.62	0.53	0.67	5.2	21.6	21.7	22.1	22.0
	1.0	0.02	0.63	0.62	0.53	0.66	4.3	21.1	21.3	21.7	21.6
barbara	0.1	0.62	0.84	0.74	0.77	0.84	25.6	28.7	27.8	26.1	29.1
	0.2	0.38	0.71	0.66	0.66	0.72	19.6	24.9	24.2	24.3	25.2
	0.3	0.26	0.64	0.61	0.59	0.64	16.0	23.5	23.1	23.3	23.7
	0.4	0.19	0.60	0.58	0.54	0.61	13.5	22.8	22.5	22.6	22.8
	0.5	0.14	0.58	0.56	0.51	0.58	11.6	22.3	22.1	22.1	22.3
	0.6	0.10	0.56	0.55	0.48	0.56	10.0	21.8	21.8	21.6	21.9
	0.7	0.08	0.54	0.54	0.44	0.54	8.7	21.5	21.5	21.0	21.6
	0.8	0.07	0.52	0.53	0.42	0.53	7.5	21.2	21.3	20.7	21.3
	0.9	0.05	0.51	0.52	0.41	0.52	6.5	20.9	21.0	20.4	21.0
	1.0	0.04	0.50	0.51	0.42	0.51	5.6	20.6	20.8	20.3	20.8
boat	0.1	0.57	0.82	0.80	0.81	0.82	25.3	30.4	29.1	30.0	30.8
	0.2	0.31	0.72	0.71	0.70	0.73	19.3	27.4	26.7	27.2	27.7
	0.3	0.19	0.66	0.66	0.65	0.67	15.8	25.8	25.4	25.7	26.0
	0.4	0.13	0.62	0.61	0.60	0.63	13.3	24.7	24.3	24.7	24.9
	0.5	0.10	0.59	0.59	0.56	0.60	11.4	23.9	23.8	23.9	24.1
	0.6	0.07	0.57	0.56	0.54	0.57	9.8	23.3	23.2	23.3	23.5
	0.7	0.06	0.55	0.55	0.51	0.56	8.4	22.8	22.7	22.8	23.0
	0.8	0.04	0.53	0.53	0.50	0.54	7.3	22.3	22.4	22.4	22.5
	0.9	0.04	0.52	0.52	0.49	0.53	6.3	21.9	22.1	22.1	22.2
	1.0	0.03	0.51	0.51	0.48	0.52	5.3	21.6	21.8	21.8	21.9
fingerprint	0.1	0.84	0.92	0.93	0.17	0.92	24.6	28.2	28.8	16.5	28.2
	0.2	0.61	0.82	0.85	0.17	0.83	18.6	24.5	25.1	16.4	24.5
	0.3	0.43	0.73	0.77	0.17	0.75	15.0	22.4	23.0	16.4	22.4
	0.4	0.31	0.66	0.70	0.17	0.68	12.5	21.0	21.5	16.4	21.1
	0.5	0.23	0.59	0.64	0.17	0.60	10.6	20.0	20.6	16.4	20.0
	0.6	0.18	0.53	0.61	0.17	0.55	9.0	19.2	20.0	16.3	19.3
	0.7	0.14	0.49	0.55	0.16	0.49	7.7	18.6	19.2	16.3	18.6
	0.8	0.11	0.46	0.49	0.16	0.42	6.5	18.2	18.5	16.3	18.0
	0.9	0.09	0.43	0.47	0.16	0.42	5.5	17.9	18.2	16.2	17.8
	1.0	0.08	0.40	0.42	0.15	0.38	4.6	17.6	17.7	16.1	17.4

Table 4.6: Comparison of anisotropic diffusion denoising and BM3D method. We also test the sensitivity of BM3D with respect to the input noise levels. The input noise levels are the exact noise level, n_0, an underestimated value, $0.7 \times n_0$, and an overestimated value, $1.3 \times n_0$.

images	noise level	MSSIM					PSNR				
		noisy	BoxMG	BM3D			noisy	BoxMG	BM3D		
				n_0	$0.7n_0$	$1.3n_0$			n_0	$0.7n_0$	$1.3n_0$
cameraman	0.1	0.53	0.84	0.91	0.83	0.88	25.5	29.8	32.3	30.8	31.4
	0.2	0.31	0.77	0.84	0.65	0.83	19.5	26.7	29.0	26.7	28.4
	0.3	0.22	0.72	0.80	0.52	0.79	16.0	24.9	27.1	24.1	26.2
	0.4	0.16	0.68	0.77	0.38	0.76	13.5	23.6	25.8	20.9	25.1
	0.5	0.13	0.65	0.74	0.40	0.74	11.5	22.6	24.9	21.9	24.2
	0.6	0.10	0.63	0.72	0.35	0.71	10.0	21.9	24.0	20.9	23.5
	0.7	0.08	0.61	0.69	0.32	0.69	8.6	21.2	23.3	20.0	22.8
	0.8	0.07	0.59	0.67	0.28	0.67	7.5	20.7	22.7	19.3	22.3
	0.9	0.06	0.58	0.65	0.26	0.65	6.4	20.3	22.2	18.6	21.8
	1.0	0.05	0.57	0.62	0.22	0.63	5.5	20.0	21.6	17.5	21.2
house	0.1	0.45	0.84	0.88	0.79	0.87	24.3	31.5	34.6	31.5	34.1
	0.2	0.23	0.78	0.84	0.59	0.83	18.3	28.2	31.7	27.1	31.2
	0.3	0.15	0.74	0.81	0.42	0.81	14.8	26.3	29.8	23.4	29.4
	0.4	0.10	0.71	0.78	0.39	0.79	12.3	25.0	28.3	23.2	28.0
	0.5	0.07	0.69	0.76	0.33	0.76	10.3	24.1	27.1	21.7	26.7
	0.6	0.05	0.67	0.73	0.28	0.74	8.8	23.3	26.0	20.6	25.8
	0.7	0.04	0.65	0.70	0.25	0.72	7.4	22.6	25.2	19.7	25.0
	0.8	0.03	0.64	0.68	0.22	0.70	6.3	22.1	24.4	18.9	24.4
	0.9	0.03	0.63	0.63	0.18	0.66	5.2	21.6	23.3	17.3	23.2
	1.0	0.02	0.63	0.53	0.14	0.56	4.3	21.1	21.9	15.5	22.0
barbara	0.1	0.62	0.84	0.93	0.86	0.92	25.6	28.7	33.5	30.8	33.1
	0.2	0.39	0.71	0.88	0.71	0.87	19.6	24.9	30.2	26.6	29.7
	0.3	0.26	0.64	0.83	0.57	0.81	16.0	23.5	28.2	23.7	27.6
	0.4	0.19	0.60	0.78	0.41	0.76	13.5	22.8	26.7	20.4	26.1
	0.5	0.14	0.58	0.74	0.45	0.72	11.6	22.3	25.5	21.6	25.0
	0.6	0.11	0.56	0.70	0.39	0.68	10.0	21.8	24.6	20.6	24.1
	0.7	0.08	0.54	0.66	0.35	0.64	8.7	21.5	23.8	19.7	23.3
	0.8	0.07	0.52	0.63	0.31	0.61	7.5	21.2	23.1	19.0	22.6
	0.9	0.05	0.51	0.59	0.28	0.58	6.5	20.9	22.4	18.4	22.1
	1.0	0.04	0.50	0.54	0.24	0.54	5.6	20.6	21.6	17.4	21.4
boat	0.1	0.57	0.82	0.86	0.80	0.84	25.3	30.4	32.5	30.5	31.8
	0.2	0.31	0.72	0.79	0.64	0.76	19.3	27.4	29.4	26.6	28.8
	0.3	0.20	0.66	0.73	0.50	0.70	15.8	25.8	27.5	23.9	26.9
	0.4	0.13	0.62	0.68	0.35	0.66	13.3	24.7	26.2	20.6	25.8
	0.5	0.10	0.59	0.65	0.37	0.64	11.4	23.9	25.3	21.7	25.0
	0.6	0.07	0.57	0.62	0.31	0.61	9.8	23.3	24.6	20.6	24.4
	0.7	0.06	0.55	0.60	0.27	0.59	8.4	22.8	24.0	19.8	23.8
	0.8	0.04	0.53	0.57	0.24	0.57	7.3	22.3	23.5	19.1	23.3
	0.9	0.04	0.52	0.55	0.21	0.56	6.3	21.9	23.0	18.4	22.9
	1.0	0.03	0.51	0.52	0.18	0.53	5.3	21.6	22.4	17.2	22.3
fingerprint	0.1	0.85	0.92	0.95	0.92	0.94	24.6	28.2	30.3	28.1	29.8
	0.2	0.62	0.82	0.89	0.83	0.87	18.6	24.5	26.8	24.0	26.3
	0.3	0.44	0.73	0.84	0.71	0.82	15.0	22.4	25.0	20.8	24.4
	0.4	0.32	0.66	0.80	0.70	0.77	12.5	21.0	23.7	20.8	23.3
	0.5	0.24	0.59	0.77	0.64	0.74	10.6	20.0	22.8	19.6	22.5
	0.6	0.18	0.53	0.73	0.59	0.71	9.0	19.2	22.0	18.7	21.8
	0.7	0.14	0.49	0.70	0.54	0.67	7.7	18.6	21.3	18.0	21.2
	0.8	0.11	0.46	0.66	0.50	0.63	6.5	18.2	20.7	17.3	20.5
	0.9	0.09	0.43	0.60	0.43	0.54	5.5	17.9	19.7	16.1	19.3
	1.0	0.08	0.40	0.49	0.32	0.44	4.6	17.6	18.3	14.4	18.0