

An Efficient Operator-Splitting Method for Noise Removal in Images

D. Krishnan^{1,*}, P. Lin¹ and X.-C. Tai²

¹ *Department of Mathematics, National University of Singapore, Singapore 117453.*

² *Department of Mathematics, University of Bergen, Johannes Brunsgate 12, Bergen, N-5008, Norway.*

Received 5 October 2005; Accepted (in revised version) 5 March 2006

Abstract. In this work, noise removal in digital images is investigated. The importance of this problem lies in the fact that removal of noise is a necessary pre-processing step for other image processing tasks such as edge detection, image segmentation, image compression, classification problems, image registration etc. A number of different approaches have been proposed in the literature. In this work, a non-linear PDE-based algorithm is developed based on the ideas proposed by Lysaker, Osher and Tai [*IEEE Trans. Image Process.*, 13 (2004), 1345-1357]. This algorithm consists of two steps: flow field smoothing of the normal vectors, followed by image reconstruction. We propose a finite-difference based additive operator-splitting method that allows for much larger time-steps. This results in an efficient method for noise-removal that is shown to have good visual results. The energy is studied as an objective measure of the algorithm performance.

Key words: Noise removal; nonlinear PDEs; additive operator splitting (AOS).

1 Introduction

In this work, removal of additive, zero-mean noise in digital images is investigated. We use the ideas proposed in [1], based on a TV-norm approach. This results in two nonlinear partial differential equations. The first of these equations is the smoothing of the flow-field (normal field) of the original image. The second equation reconstructs a noise-reduced image from the smoothed flow-field. This results in an efficient method for noise-removal that has good results.

*Correspondence to: D. Krishnan, Department of Mathematics, National University of Singapore, Singapore 117453. Email: dilipk@nus.edu.sg

The contributions of this paper are as follows: a more efficient scheme for flow-field smoothing is developed; an additive operator splitting (AOS) method ([2, 15, 16]) is employed to further improve the efficiency of the flow-field smoothing; a comparison of the AOS method and explicit methods are done in terms of numerical performance; and lastly, the use of the energy is suggested as an objective measure of the performance of noise-reduction algorithms based on energy minimization.

Let d be a digital image defined on a two-dimensional region Ω . Let (x, y) denote the position of a single pixel. $d(x, y)$ is the grey-level value associated with the pixel (x, y) . The noise model is assumed to be zero-mean and additive, denoted as $\eta(x, y)$. The observed image values $d_0(x, y)$ are

$$d_0(x, y) = d(x, y) + \eta(x, y). \quad (1.1)$$

The problem is to recover the (unknown) true image $d(x, y)$ from the given observations $d_0(x, y)$.

The visually annoying parts of the noise usually belong in the higher frequency regions of the spectrum. A lot of noise can be effectively filtered by a lowpass filter. However, this will remove the true high-frequency components of the image such as edges and texture. Isotropic filters suffer from this problem. The challenge is to retain as much of the true high-frequency information as possible while reducing the perceived noise levels in the image.

The Total Variation (TV) norm based filters proposed in [5], have been shown to be quite effective in removing noise without causing excessive smoothing of the edges. The original formulation of this filter is to obtain $d(x, y)$ as a solution of the constrained optimization problem

$$\inf_d \int_{\Omega} |\nabla d| \, dx \quad \text{subject to} \quad \int_{\Omega} |d - d_0|^2 \, dx = \sigma^2, \quad (1.2)$$

where σ^2 represents the noise level. The resulting Euler-Lagrange PDE to be solved in this case is

$$-\nabla \cdot \left(\frac{\nabla d}{|\nabla d|} \right) + \mu(d - d_0) = 0. \quad (1.3)$$

However, it is well known that the TV norm filter has the disadvantage of a *stair-case effect*: smooth functions get transformed into piecewise constant functions. This lends an undesirable blocky effect to the smoothed image. In [1], it is proposed to modify the equation (1.2). Instead of minimizing the TV norm of d , it is proposed in [1] to minimize the TV norm of $\nabla d/|\nabla d|$, giving the following equation

$$\inf_d \int_{\Omega} \left| \nabla \frac{\nabla d}{|\nabla d|} \right| \, dx \quad \text{subject to} \quad \int_{\Omega} |d - d_0|^2 \, dx = \sigma^2, \quad (1.4)$$

$\vec{n} = \nabla d/|\nabla d|$ is called the normal field of the image. The fourth-order Euler-Lagrange equation that results from directly minimizing this functional is difficult to solve numerically in a stable manner. Therefore, in [1], a re-formulation of this equation is done. The

above equation is split into two steps. The first step involves the smoothing of the normal vectors $\vec{n}_0 = \nabla d_0 / |\nabla d_0|$. The second step involves reconstructing an image whose normal field fits the smoothed normals according to the noise variance levels σ^2 . It is shown in [1] that this two-step procedure provides visual results that are superior to that of the original TV norm smoothing scheme (1.2).

This paper is organized as follows. In Section 2, several issues on flow field smoothing are discussed. A new model is proposed which converts the Euler-Lagrange equation to a formulation based on angles. It will be demonstrated that this approach is more efficient than the use of the original formulation. A semi-implicit AOS scheme [2, 15, 16] for the numerical solution of the problem is also presented. In Section 3, an equation for the reconstruction of the image from the previously smoothed normal field is presented, along with an AOS scheme. In Section 4, computational results are presented. The performance of the AOS scheme is compared against that of the explicit scheme. It is also shown that the energy is a good measure of the performance of the algorithms, and this can be used as an objective performance measure.

2 Flow field smoothing

For a given image d , the unit normal field \vec{n} is given by $\vec{n} = \nabla d / |\nabla d|$. Let $\vec{n}_0 = \nabla d_0 / |\nabla d_0|$ denote the normal flow-field of the observed noisy image d_0 . In numerical implementation, in areas of low gradient magnitude ($|\nabla d| \ll 1$), a small constant is added to the evaluation of $|\nabla d|$, i.e. $|\nabla d|$ is replaced by $\sqrt{|\nabla d|^2 + \epsilon}$. The TV norm minimization to smooth the flow field is

$$\inf_{|\vec{n}|=1} \left\{ \int_{\Omega} |\nabla \vec{n}| + \frac{\lambda}{2} |\vec{n} - \vec{n}_0|^2 dx \right\}, \quad (2.1)$$

where λ is a constant that balances smoothing and fidelity to the observed flow field. In [1], it is suggested that λ be set between 1 and 10. The value is determined by experimentation. Setting λ to 0 is equivalent to smoothing using the original TV norm model as proposed in [5].

The unit vector constraint makes this a non-convex minimization problem. This may be approached directly, as is done in [4], or indirectly through the use of either a projection method or a penalty method, similar to [14]. The direct approach involves splitting the vector \vec{n} into two components u and v corresponding to the x and y directions of the unit vector. Adding the unity constraint $u^2 + v^2 = 1$ and an artificial time variable results in the following set of PDEs to be solved

$$\frac{\partial u}{\partial t} = -v \operatorname{div} \left(\frac{u \nabla v - v \nabla u}{|u \nabla v - v \nabla u|} \right) + \lambda u [v u_0 - u v_0], \quad (2.2)$$

$$\frac{\partial v}{\partial t} = u \operatorname{div} \left(\frac{u \nabla v - v \nabla u}{|u \nabla v - v \nabla u|} \right) - \lambda u [v u_0 - u v_0], \quad (2.3)$$

where $\vec{n}_0 = (u_0, v_0)$. We shall use a new formulation here that is more efficient in terms of the number of computations involved per iteration. This is done by transforming the problem into polar co-ordinates. Let $\vec{n} = (\cos \theta, \sin \theta)$. Then it can be shown [1] that $|\nabla \vec{n}| = |\nabla \theta|$. Let $\vec{n}_0 = (\cos \theta^0, \sin \theta^0)$. Then

$$|\vec{n} - \vec{n}_0|^2 = (\cos \theta - \cos \theta^0)^2 + (\sin \theta - \sin \theta^0)^2.$$

Simplifying the right-hand side gives $|\vec{n} - \vec{n}_0|^2 = 2(1 - \cos(\theta - \theta^0))$. Then (2.1) can be re-written as

$$\inf_{\theta} \int_{\Omega} |\nabla \theta| + \lambda(1 - \cos(\theta - \theta^0)) \, dx. \quad (2.4)$$

The energy is the function

$$E(\theta) = \int_{\Omega} |\nabla \theta| + \lambda(1 - \cos(\theta - \theta^0)) \, dx. \quad (2.5)$$

The Euler-Lagrange equation corresponding to the above problem is

$$-\nabla \cdot \left\{ \frac{\nabla \theta}{|\nabla \theta|} \right\} + \lambda \sin(\theta - \theta^0) = 0. \quad (2.6)$$

With an artificial time-discretization term added, the following equation is solved to steady-state:

$$\theta_t = \nabla \cdot \left\{ \frac{\nabla \theta}{|\nabla \theta|} \right\} - \lambda \sin(\theta - \theta^0). \quad (2.7)$$

Boundary conditions are Neumann, $\nabla \theta \cdot \vec{\nu} = 0$, where $\vec{\nu}$ is the outward normal at the image boundaries. Let $\Delta_{\mp}^x \theta_{i,j} = \mp(\theta_{i\mp 1,j} - \theta_{i,j})$ and $\Delta_{\mp}^y \theta_{i,j} = \mp(\theta_{i,j\mp 1} - \theta_{i,j})$. The numerical approximations for first derivatives are

$$\theta_x \approx \Delta_-^x \theta_{i,j}^n, \quad \theta_y \approx \Delta_-^y \theta_{i,j}^n. \quad (2.8)$$

To compute an approximation to $|\nabla \theta|$, central derivatives are first defined by

$$\Delta_0^x \theta_{i,j}^n = \frac{1}{4} (\Delta_-^x \theta_{i,j}^n + \Delta_-^x \theta_{i+1,j}^n + \Delta_-^x \theta_{i,j+1}^n + \Delta_-^x \theta_{i+1,j+1}^n), \quad (2.9a)$$

$$\Delta_0^y \theta_{i,j}^n = \frac{1}{4} (\Delta_-^y \theta_{i,j}^n + \Delta_-^y \theta_{i+1,j}^n + \Delta_-^y \theta_{i,j+1}^n + \Delta_-^y \theta_{i+1,j+1}^n). \quad (2.9b)$$

Then $|\nabla \theta|$ is approximated by

$$|\nabla \theta|_{i,j;x}^n = ((\Delta_-^x \theta_{i,j}^n)^2 + (\Delta_0^y \theta_{i,j}^n)^2)^{\frac{1}{2}}, \quad |\nabla \theta|_{i,j;y}^n = ((\Delta_-^y \theta_{i,j}^n)^2 + (\Delta_0^x \theta_{i,j}^n)^2)^{\frac{1}{2}}. \quad (2.10)$$

An explicit numerical scheme to solve (2.7) is

$$\frac{\theta_{i,j}^{n+1} - \theta_{i,j}^n}{\Delta t} = \frac{1}{h} \Delta_+^x \left(\frac{\Delta_-^x \theta_{i,j}^n}{|\nabla \theta|_{i,j;x}^n} \right) + \frac{1}{h} \Delta_+^y \left(\frac{\Delta_-^y \theta_{i,j}^n}{|\nabla \theta|_{i,j;y}^n} \right) - \lambda \sin(\theta_{i,j}^n - \theta_{i,j}^0). \quad (2.11)$$

An additive spatial operator splitting (AOS) can be used for a more efficient semi-implicit numerical scheme. The AOS gives the semi-implicit scheme (c.f. [2, 15, 16])

$$d^{m+1} = \frac{1}{m} \sum_{l=1}^m [I - m\tau A_l(d^m)]^{-1} d^m, \tag{2.12}$$

where I is the identity matrix and A contains the coefficients corresponding to the diffusivity operator in (2.7). m is the problem dimension, in our case $m = 2$. This algorithm was first proposed in Lu, Neittaanmäki and Tai [15, 16]. It was discovered independently later in [2] and used in a different context for image processing. This scheme splits the numerical scheme along the axes and combines the results additively. The resulting scheme is tri-diagonal along each axis (due to the second-order diffusion operator). Therefore, the individual split models can be solved efficiently and are simple to implement. Under appropriate restrictions on the matrix A (which are satisfied by our current model), this semi-implicit scheme is numerically stable and satisfies a maximum principle [3]. With AOS splitting, the semi-implicit formulation of equation (2.11) can be re-written as

$$\frac{\bar{\theta}_{i,j}^{n+\frac{1}{2}} - \theta_{i,j}^n}{\Delta t} = \frac{1}{h} \Delta_+^x \left(\frac{\Delta_-^x \bar{\theta}_{i,j}^{n+\frac{1}{2}}}{|\nabla \theta_{i,j;x}^n|} \right) - \lambda \sin(\theta_{i,j}^n - \theta_{i,j}^0), \tag{2.13a}$$

$$\frac{\hat{\theta}_{i,j}^{n+\frac{1}{2}} - \theta_{i,j}^n}{\Delta t} = \frac{1}{h} \Delta_+^y \left(\frac{\Delta_-^y \hat{\theta}_{i,j}^{n+\frac{1}{2}}}{|\nabla \theta_{i,j;y}^n|} \right) - \lambda \sin(\theta_{i,j}^n - \theta_{i,j}^0), \tag{2.13b}$$

and

$$\theta_{i,j}^{n+1} = \frac{1}{2} \left(\bar{\theta}_{i,j}^{n+\frac{1}{2}} + \hat{\theta}_{i,j}^{n+\frac{1}{2}} \right). \tag{2.14}$$

The above tri-diagonal equations are re-arranged and iterated using a fast Thomas algorithm [9] pp. 23-25.

In [1], the Euler-Lagrange formulation of (2.1) is done by treating the two components of \vec{n} separately. More computations are introduced by the need to maintain the unit vector constraint. Comparing equation (2.7) with equations (2.2) and (2.3), shows that the new formulation requires one iteration equation instead of two coupled iteration equation. Table 1 in Section 4 confirms the speed increase that can be achieved with the new scheme. The flow-field results from both schemes are similar. Hence, in our image reconstruction results, we will only present the results with the new formulation (2.7).

3 Image reconstruction

The image reconstruction step recovers an image from the smoothed normal field that results from solving (2.1). The recovered image d satisfies

$$\frac{\nabla d}{|\nabla d|} = \vec{n} \quad \text{and} \quad \int_{\Omega} |d - \vec{d}_0|^2 dx = \sigma^2.$$

Taking the dot product of both sides of the first equation above by \vec{n} , and noting that \vec{n} is a unit vector gives $\nabla d \cdot \vec{n} = |\nabla d|$. Minimizing this over the whole domain leads to the problem to be solved

$$\inf_d \left\{ \int_{\Omega} (|\nabla d| - \nabla d \cdot \vec{n}) \, dx \right\} \quad \text{subject to} \quad \int_{\Omega} |d - d_0|^2 \, dx = \sigma^2, \quad (3.1)$$

where σ^2 is the estimated noise variance. This can be estimated using statistical techniques such in [6]. If the exact noise variance cannot be computed, then an approximate value may be used. If the input value is less than the true value, then undersmoothing will result. A larger value will result in oversmoothing. Using Lagrange multipliers for (3.1) gives the following functional

$$L(d, \mu) = \int_{\Omega} (|\nabla d| - \nabla d \cdot \vec{n}) \, dx + \frac{\mu}{2} \left(\int_{\Omega} |d - d_0|^2 - \sigma^2 \, dx \right). \quad (3.2)$$

The optimality conditions for $L(d, \mu)$ are

$$-\nabla \cdot \left(\frac{\nabla d}{|\nabla d|} - \vec{n} \right) + \mu(d - d_0) = 0 \quad \text{in } \Omega, \quad (3.3)$$

and

$$\int_{\Omega} \left(\frac{d - d_0}{\sigma} \right)^2 \, dx = 1. \quad (3.4)$$

The boundary conditions to be used are again Neumann,

$$\left(\frac{\nabla d}{|\nabla d|} - \vec{n} \right) \cdot \vec{\nu} = 0 \quad \text{on } \partial\Omega. \quad (3.5)$$

In the Numerical Results section (see below), the following portion of the Lagrange functional will be used as a measure of the quality of the reconstruction

$$E(d) = \left\{ \int_{\Omega} (|\nabla d| - \nabla d \cdot \vec{n}) \, dx \right\}. \quad (3.6)$$

From (3.3), multiplying the equation by $d - d_0$ and then integrating over Ω gives

$$\int_{\Omega} \nabla \cdot \left(\frac{\nabla d}{|\nabla d|} - \vec{n} \right) (d - d_0) \, dx = \mu \int_{\Omega} (d - d_0)^2 \, dx.$$

From (3.1) this simplifies to

$$\int_{\Omega} \nabla \cdot \left(\frac{\nabla d}{|\nabla d|} - \vec{n} \right) (d - d_0) \, dx = \mu\sigma^2. \quad (3.7)$$

By the 2-dimensional divergence theorem,

$$\int_{\Omega} \nabla \cdot \left(\frac{\nabla d}{|\nabla d|} - \vec{n} \right) (d - d_0) + \left(\frac{\nabla d}{|\nabla d|} - \vec{n} \right) \cdot \nabla (d - d_0) \, dx = \int_{\partial\Omega} \left(\frac{\nabla d}{|\nabla d|} - \vec{n} \right) \cdot \vec{\nu},$$

where $\vec{\nu}$ is the outward normal on the boundary $\partial\Omega$. From (3.5), the right hand side above is 0. Therefore, we can get the following formula for μ

$$\mu = -\frac{1}{\sigma^2} \int_{\Omega} \left(\frac{\nabla d}{|\nabla d|} - \vec{n} \right) \cdot \nabla(d - d_0) \, dx. \tag{3.8}$$

Introducing a time variable t for (3.3) gives

$$d_t = \nabla \cdot \left(\frac{\nabla d}{|\nabla d|} - \vec{n} \right) - \mu(d - d_0) \tag{3.9}$$

cf. [1], which is solved to steady-state. The above equations are discretized as follows:

$$\begin{aligned} d_x &\approx \Delta_-^x d_{i,j}^n, & d_y &\approx \Delta_-^y d_{i,j}^n, \\ \Delta_0^x d_{i,j}^n &= \frac{1}{4} (\Delta_-^x d_{i,j}^n + \Delta_-^x d_{i+1,j}^n + \Delta_-^x d_{i,j+1}^n + \Delta_-^x d_{i+1,j+1}^n), \\ \Delta_0^y d_{i,j}^n &= \frac{1}{4} (\Delta_-^y d_{i,j}^n + \Delta_-^y d_{i+1,j}^n + \Delta_-^y d_{i,j+1}^n + \Delta_-^y d_{i+1,j+1}^n), \\ |\nabla d_{i,j;x}^n| &= ((\Delta_-^x d_{i,j}^n)^2 + (\Delta_0^y d_{i,j}^n)^2)^{\frac{1}{2}}, & |\nabla d_{i,j;y}^n| &= ((\Delta_-^y d_{i,j}^n)^2 + (\Delta_0^x d_{i,j}^n)^2)^{\frac{1}{2}}, \end{aligned} \tag{3.10}$$

and

$$\mu^n = -\frac{h}{\sigma^2} \sum_{i,j} \left[\left(\frac{\Delta_-^x d_{i,j}^n}{|\nabla d_{i,j;x}^n|} - u_{i,j} \right) \times \Delta_-^x (d_{i,j}^n - d_{i,j}^0) + \left(\frac{\Delta_-^y d_{i,j}^n}{|\nabla d_{i,j;y}^n|} - v_{i,j} \right) \times \Delta_-^y (d_{i,j}^n - d_{i,j}^0) \right].$$

The explicit reconstruction equation is

$$\frac{d_{i,j}^{n+1} - d_{i,j}^n}{\Delta t} = \frac{1}{h} \Delta_+^x \left(\frac{\Delta_-^x d_{i,j}^n}{|\nabla d_{i,j;x}^n|} - u_{i,j} \right) + \frac{1}{h} \Delta_+^y \left(\frac{\Delta_-^y d_{i,j}^n}{|\nabla d_{i,j;y}^n|} - v_{i,j} \right) - \mu^n (d_{i,j}^n - d_{i,j}^0). \tag{3.11}$$

If the values for $u_{i,j}$ and $v_{i,j}$ are set to 0, then the iterations above correspond to the TV norm reconstruction (1.2). As before, Δt is the time step, and h is the space mesh size. An additive operator splitting on the above equation, similar to (2.13)-(2.14) gives

$$\frac{\bar{d}_{i,j}^{n+\frac{1}{2}} - d_{i,j}^n}{\Delta t} = \frac{1}{h} \Delta_+^x \left(\frac{\Delta_-^x \bar{d}_{i,j}^{n+\frac{1}{2}}}{|\nabla d_{i,j;x}^n|} \right) - \mu(\bar{d}_{i,j}^{n+\frac{1}{2}} - d_{i,j}^0) - \frac{1}{h} (\Delta_+^x u_{i,j} + \Delta_+^y v_{i,j}), \tag{3.12a}$$

$$\frac{\hat{d}_{i,j}^{n+\frac{1}{2}} - d_{i,j}^n}{\Delta t} = \frac{1}{h} \Delta_+^y \left(\frac{\Delta_-^y \hat{d}_{i,j}^{n+\frac{1}{2}}}{|\nabla d_{i,j;y}^n|} \right) - \mu(\hat{d}_{i,j}^{n+\frac{1}{2}} - d_{i,j}^0) - \frac{1}{h} (\Delta_+^x u_{i,j} + \Delta_+^y v_{i,j}), \tag{3.12b}$$

and

$$d_{i,j}^{n+1} = \frac{1}{2} \left(\bar{d}_{i,j}^{n+\frac{1}{2}} + \hat{d}_{i,j}^{n+\frac{1}{2}} \right). \tag{3.13}$$

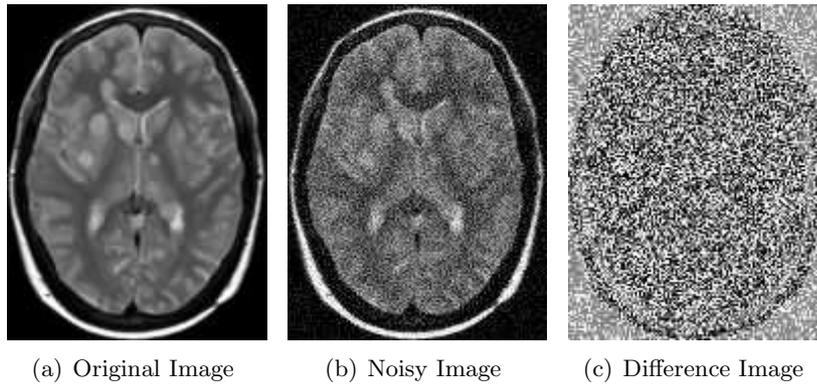


Figure 1: Brain MR image of size 167×128 with $\text{SNR} = 25$.

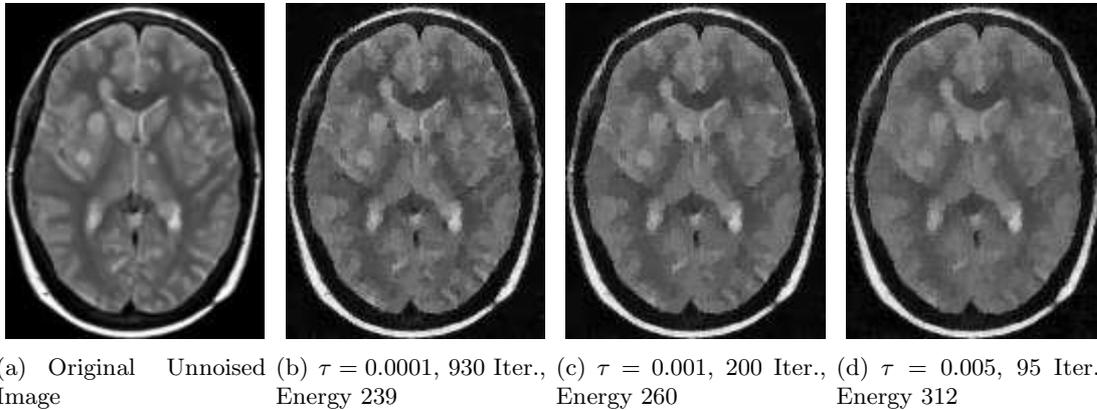


Figure 2: Comparison of output for different stable energies.

4 Numerical results

Fig. 1 shows zero-mean noise added to a test image, a brain Magnetic Resonance (MR) image. Table 1 shows the speed gains that are made in the flow-field smoothing by using the new formulation (2.7) (θ method) in place of the formulation (2.2) and (2.3) ((u, v) method). For 100 iterations, the total CPU time for the (u, v) method is more than three times that of the θ method. In addition, the (u, v) method takes longer to converge than the θ method. We have not investigated the reason for this, but this further increases the efficiency gains brought about by the θ method.

To test the performance of the AOS and explicit schemes for flow-field smoothing and image reconstruction, it is proposed to use the energy functionals (2.5) and (3.6) as a measure of the quality of the results. The energies represent the quantity that we want to minimize in the first place. It is therefore natural to expect that a lower energy will give a result of higher quality. Fig. 2 shows the results of processing the brain image with

Table 1: Efficiency comparison.

Method.	CPU Time
θ , 100 iter.	12.919
(u, v) , 100 iter.	44.2030

Table 2: Explicit scheme energies for various values of λ .

Time Step	$\lambda = 0$			$\lambda = 2$			$\lambda = 4$		
	Eng.(11)	Eng.(4)	Iter.	Eng.(11)	Eng.(4)	Iter.	Eng.(11)	Eng.(4)	Iter.
0.001	963	963	2000	37343	9573	2000	36728	18448	2000
0.01	336	336	680	36394	8361	2000	37160	18368	2000
0.1	2120	2120	250	35427	9988	2000	37179	18383	2000
1.0	7339	7339	2000	72116	24948	2000	141521	59727	2000
2.0	35172	35172	2000	94367	51460	2000	195543	112598	2000

Table 3: AOS scheme energies for various values of λ .

Time Step	$\lambda = 0$			$\lambda = 2$			$\lambda = 4$		
	Eng.(11)	Eng.(4)	Iter.	Eng.(11)	Eng.(4)	Iter.	Eng.(11)	Eng.(4)	Iter.
0.001	124	124	2000	41476	4724	140	51039	15128	2000
0.01	0	0	355	41647	1429	255	48792	16060	2000
0.1	0	0	65	34834	11421	285	44481	16824	280
1.0	0	0	30	30789	15950	175	81957	22370	2000
2.0	0	0	25	47753	17589	2000	133351	48281	2000

different time steps for the explicit scheme (3.11) for the same input normal field \vec{n} . The time step, iterations and final steady-state energies are also shown below the figures. The iterations were terminated when the change in energy falls below 10^{-1} . This is usually a reasonable criterion for termination, since the energies are in the order of hundreds or thousands for a typical image. It is seen that a final lower energy leads to a visually superior image. Similar results are observed for our other test cases. Therefore, we suggest that the energy can be used as an objective measure of the quality of a numerical scheme for a particular energy minimization problem.

Now, we compare the results of flow-field smoothing of Section 2 using the AOS scheme (2.13)- (2.14) to the explicit scheme of (2.11). The comparisons are done for the brain image of Fig. 1. As before, the iterations are terminated when the change in energy between 2 consecutive iterations falls below 10^{-1} . Tables 2 and 3 below show the final energies and number of iterations for the two schemes at various time steps and with varying value of λ . We compare the energies (2.5) and (1.4). The reason to compare (2.5) is because this is the problem that we are solving with the explicit and AOS schemes. However, (1.4) is the problem that we were trying to solve in the first place, and so a comparison of this energy is also warranted. The maximum number of iterations is 2000, at which point we always terminate since the scheme would be impractical for more iterations. The explicit scheme is unstable for time steps larger than 0.1, as can be seen from the drastic increase

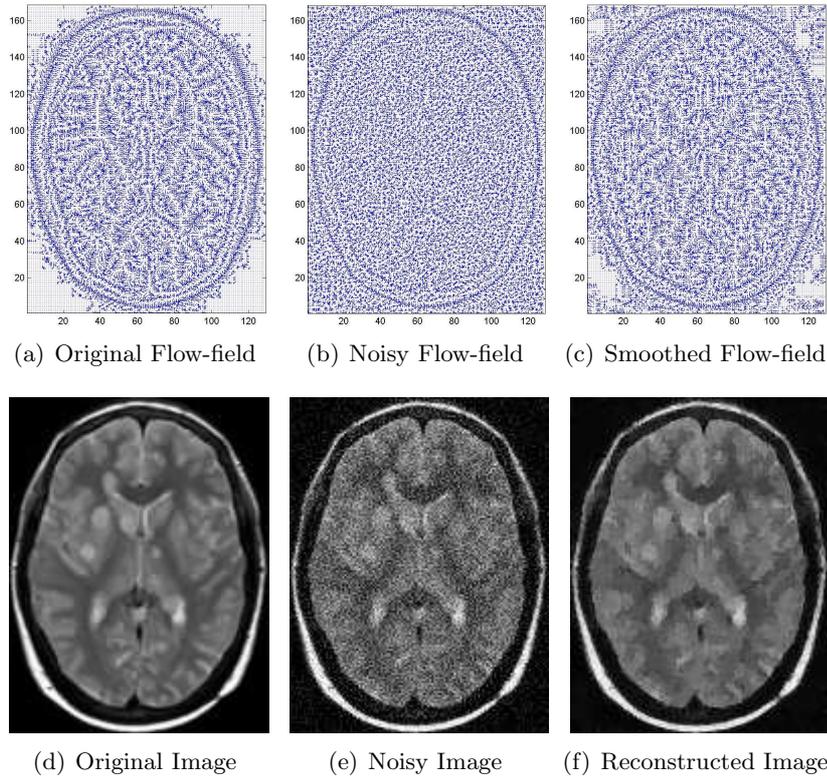


Figure 3: Flow-field smoothing results for the brain image of Fig. 1.

in energy. In all experiments, we always set the space mesh size to 1, according to common practice. It can be seen that for smaller values of λ , the AOS method is superior in terms of lower energies reached at steady-state. For $\lambda = 4$, the explicit scheme is superior for the energies (2.5), but the AOS method converges in one instance in much fewer iterations. In the original paper [1], it was suggested that the value of λ was set to 1 or 2 generally. With these values, we can use the AOS method for the flow-field smoothing.

To show the superior performance of the new model, in Table 4, the relative errors are shown for smoothing with the original TV norm equation (1.2) and the new flow-field smoothing and reconstruction model. The relative error is given by comparing the smoothed image ds with the original, undergraded test image d , and is given by $\|ds\|/\|d\|$. It is seen that the new model provides a superior reconstruction. This is in agreement with the results given in [1].

The equations (2.7) and (3.9) are similar in their formulations, except for the presence of the additional $\nabla \cdot (\vec{n})$ term and the fact that the fidelity term parameter μ is a function of time, instead of a constant like λ . Our numerical investigations reveal that the behaviour of the AOS and explicit numerical schemes is similar for both these equations. As the value of the fidelity term parameter (λ or μ) increases, the explicit scheme provides a

Table 4: Relative errors for (1.2) and the new model.

Image	Original TV	New Model
Brain	0.041255	0.041096
Peppers	0.015316	0.014054

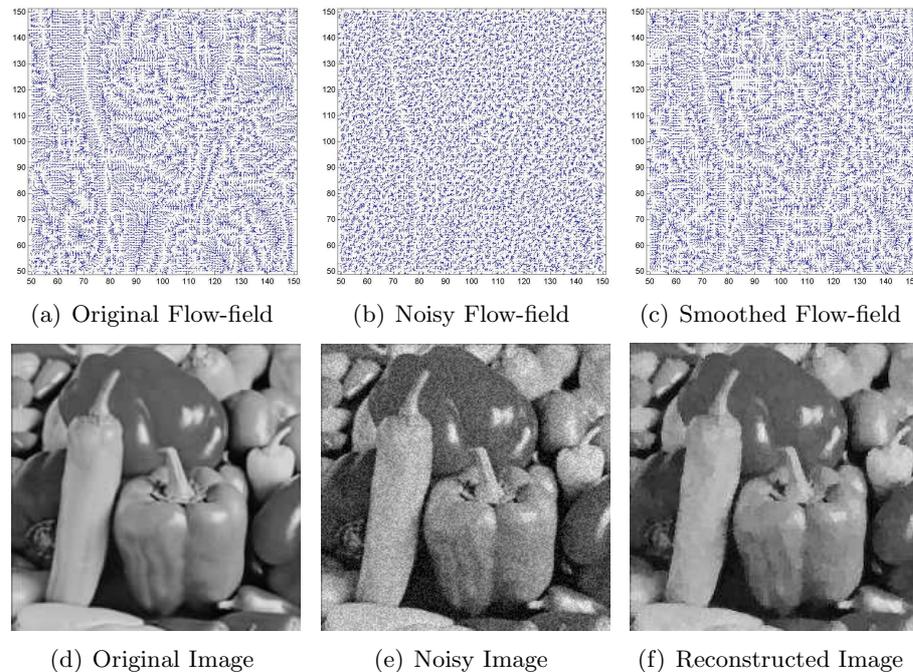


Figure 4: Flow-field smoothing results for the Peppers image with SNR 60.

superior result in terms of lower energy. In case of the reconstruction, the value of μ is usually of a higher value than 4. The explicit numerical scheme therefore provides a superior result. We have also tested the result with a fully implicit scheme, and found that this provides the lowest energy stable result with a larger time step, than either the explicit or the AOS schemes. Therefore, we hypothesize that the splitting error of the semi-implicit scheme is the cause for the poorer performance at higher absolute values of the fidelity term parameter.

Below are shown results of the processing of noisy original data. In Fig. 3, the brain image of Fig. 1 is processed. The value of λ is fixed at 2, and the AOS scheme is used with time step 1 for the processing. This offers a reasonable trade-off between smoothing and fidelity to the original data. It is seen that the smoothed normal vectors are closer in alignment with the vectors of the original (noise-free) image, especially at the major edges. The number of iterations was 175. The explicit scheme with time step 0.0001 is used for the image reconstruction. The number of iterations was 930. It is seen that the

noise has been removed to a large extent and the edges are well-preserved.

Fig. 4 shows the results of noise removal for the Peppers image with SNR 60. The value of λ was set to 2. The time steps used were the same as for the brain image, i.e. 1 for the flow-field smoothing and 0.0001 for the image reconstruction. The minimum energies reached were 61701 for the smoothing and 370 for the reconstruction. The number of iterations was 165 for the flow-field smoothing and 1055 for the reconstruction.

References

- [1] M. Lysaker, S. Osher and X.-C. Tai, Noise removal using smoothed normals and surface fitting, *IEEE T. Image Process.*, 13(10) (2004), 1345-1357.
- [2] J. Weickert, B. M. ter Haar Romeny and M. A. Viergever, Efficient and reliable schemes for nonlinear diffusion filtering, *IEEE T. Image Process.*, 7(3) (1998), 398-410.
- [3] J. Weickert, *Anisotropic Diffusion in Image Processing*, Stuttgart, Germany, Teubner, 1998.
- [4] L. A. Vese and S. Osher, Numerical methods for P-harmonic flows and applications to image processing, *SIAM J. Numer. Anal.*, 40(6) (2002), 2085-2104.
- [5] L. I. Rudin, S. Osher and E. Fatemi, Nonlinear total variation based noise removal algorithms, *Physica D*, 60 (1992), 259-268.
- [6] P. Meer, J-M. Jolion and A. Rosenfeld, A fast parallel algorithm for blind estimation of noise variance, *IEEE T. Pattern Anal. Mach. Intell.*, 12 (1990), 216-223.
- [7] S. Osher, A. Sole and L. Vese, Image decomposition and restoration using total variation minimization and the H^{-1} norm, *Multiscale Model. Simul.*, 1(3) (2003), 349-370.
- [8] P. Perona and J. Malik, Scale space and edge detection using anisotropic diffusion, *IEEE T. Pattern Anal. Mach. Intell.*, 12 (1990), 629-639.
- [9] K. W. Morton and D. F. Mayers, *Numerical Solution of Partial Differential Equations*, Cambridge University Press, Cambridge, United Kingdom, 1994.
- [10] N. N. Yanenko, *The Method of Fractional Steps*, Springer-Verlag, Berlin, 1971.
- [11] G. H. Golub and C. F. Van Loan, *Matrix Computations*, The Johns Hopkins University Press, Third Edition, Maryland, USA, 1996.
- [12] R. A. Varga, *Iterative Matrix Analysis*, Prentice Hall, Englewood Cliffs, New Jersey, 1962.
- [13] X.-C. Tai, O. Christiansen, P. Lin and I. Skjaelaaen, A remark on the MBO scheme and some piecewise constant level set methods, submitted.
- [14] R. Glowinski, P. Lin and X.-B. Pan, An operator-splitting method for a liquid crystal model, *Comput. Phys. Commun.*, 152 (2003), 242-252.
- [15] T. Lu, P. Neittaanmäki and X.-C. Tai, A parallel splitting up method and its application to Navier-Stokes equations, *Appl. Math. Lett.*, 4(2) (1991), 25-29.
- [16] T. Lu, P. Neittaanmäki and X.-C. Tai, A parallel splitting up method for partial differential equations and its application to Navier-Stokes equations, *RAIRO Math. Model. Numer. Anal.*, 26 (1992), 673-708.