Published in final edited form as:
Magn Reson Imaging. 2010 February ; 28(2): 200–211. doi:10.1016/j.mri.2009.10.001.

Efficient Anisotropic Filtering of Diffusion Tensor Images

Qing Xu[1,2], **Adam W. Anderson**[1,3], **John C. Gore**[1,3], and **Zhaohua Ding**[1,2,3]

[1] Vanderbilt University Institute of Imaging Science, Vanderbilt University

[2] Department of Electrical Engineering and Computer Science, Vanderbilt University

[3] Department of Biomedical Engineering, Vanderbilt University

Abstract

To improve the accuracy of structural and architectural characterization of living tissue with diffusion tensor imaging, an efficient smoothing algorithm is presented for reducing noise in diffusion tensor images. The algorithm is based on anisotropic diffusion filtering, which allows both image detail preservation and noise reduction. However, traditional numerical schemes for anisotropic filtering have the drawback of inefficiency and inaccuracy due to their poor stability and first order time accuracy. To address this, an unconditionally stable and second order time accuracy semi-implicit Craig-Sneyd scheme is adapted in our anisotropic filtering. By using large step size, unconditional stability allows this scheme to take much fewer iterations and thus less computation time than the explicit scheme to achieve a certain degree of smoothing. Second order time accuracy makes the algorithm reduce noise more effectively than a first order scheme with the same total iteration time. Both the efficiency and effectiveness are quantitatively evaluated based on synthetic and *in vivo* human brain diffusion tensor images, and these tests demonstrate that our algorithm is an efficient and effective tool for denoising diffusion tensor images.

Keywords

Diffusion tensor images; anisotropic smoothing; semi-implicit scheme

1. Introduction

Magnetic resonance diffusion tensor imaging (DTI) has become established as a primary technique for non-invasive characterization of the structural and architectural features of living tissue [1,2]. In each voxel, DTI provides a 3×3 symmetric positive definite matrix that describes the local Brownian motion of water molecules. Eigenvalues and eigenvectors of the matrix can be exploited to characterize tissue micro-structure and architecture and assess fiber integrity based on differences in the eigenvalues or orientation of the eigenvectors [3].

As DTI is typically performed with echo-planar imaging sequences, the images acquired usually have very poor signal-to-noise ratio (SNR). High image noise has quite detrimental effects on the accuracy of assessment of diffusion anisotropy (from which tissue structure is characterized), most notably an overestimate of fractional anisotropy (FA) due to sorting bias

Corresponding Author: Zhaohua Ding, Ph.D., Vanderbilt University Institute of Imaging Science, 1161 21st Avenue South, MCN AA-1105, Nashville, TN 37232-2310, Phone: (615)322-7889, Fax: (615)322-0734, zhaohua.ding@vanderbilt.edu.

Publisher's Disclaimer: This is a PDF file of an unedited manuscript that has been accepted for publication. As a service to our customers we are providing this early version of the manuscript. The manuscript will undergo copyediting, typesetting, and review of the resulting proof before it is published in its final citable form. Please note that during the production process errors may be discovered which could affect the content, and all legal disclaimers that apply to the journal pertain.

[4] and erroneous calculations of the principal diffusion direction [5,6]. To improve SNR, it is a common practice to use time domain signal averaging; this however necessarily incurs other complications such as inter-scan registration and prolonged acquisition time which may worsen the problem of subject motion. A very attractive alternative is noise reduction by image smoothing, which offers an economical and practically feasible means of improving SNR without encumbering the image acquisition procedure.

To date, a plethora of image smoothing techniques have been proposed for reducing noise in DTI data. These include non-linear diffusion filtering [7,8], B-spline fitting [9], and more sophisticated regularization methods based on a Markovian model [10], stochastic relaxation [11], variational principles [12,13], differential geometry [14–17], and a maximum a *posteriori* approach [18]. This repository of smoothing techniques, however, have not established their practical utility due, in part, to the somewhat time-consuming numerical implementation especially given the fact that computation complexity increases with the number of weighting directions, or to a lack of rigorous validation with *in vivo* DTI data to prove their practical value.

Previously we proposed to reduce noise in DTI data by anisotropic filtering of diffusion weighted images [19]. This method improves the nonlinear diffusion filtering by Parker et al. [7] in that the spatially varying diffusion coefficient is replaced with a diffusion tensor, which allows direction dependent smoothing for each voxel. In a typical implementation of diffusion filtering, an explicit finite difference scheme is used to solve the relevant partial differential equation (PDE). Explicit schemes are only stable for very small time step sizes, which imposes a significant restriction on their computational efficiency. Furthermore, explicit schemes are limited to first order accuracy in time, which means the filtering results are less accurate than those from higher order schemes.

The issue of computational efficiency is often addressed by using unconditionally stable semi-implicit schemes, which solve systems of linear equations in each iteration. The linear system is usually split into a sequence of simple ones, each of which can be solved by the efficient Thomas algorithm [20]. For example, Weickert [21] proposed a reliable and efficient additive operator splitting (AOS) scheme that allows larger time step sizes in nonlinear diffusion filtering. The AOS scheme however has the drawback of first order time accuracy only. Barash [22] used an additive and multiplicative operator splitting (AMOS) algorithm, which resembles the classical alternating direction implicit schemes [20]. Although AMOS has second order accuracy in time, it only works for diffusion filters with a scalar diffusion coefficient. In this work we employ an unconditionally stable and second order accurate Craig-Sneyd scheme [23], which improves the anisotropic filtering with respect to both accuracy and efficiency.

In the following, the anisotropic smoothing technique we proposed is first briefly summarized. A semi-implicit Craig-Sneyd scheme for improving both the accuracy and time efficiency of diffusion filtering is then described, followed by detailed implementation procedures for efficient anisotropic smoothing. Comprehensive validations with DTI data synthesized with physiological diffusion parameters and acquired *in vivo* from a healthy human brain are presented. Explicit and semi-implicit implementations of the anisotropic smoothing algorithm are compared in terms of both the efficiency and effectiveness in restoring the PDD from noise corrupted data. Finally, major contributions of this work are summarized and some technical limitations are discussed so as to guide future research along this line.

2. Methods

2.1. Anisotropic reduction of noise in diffusion tensor images

Noise in DTI data can be reduced by smoothing diffusion weighted images (DWI) from which diffusion tensors are derived. A common smoothing approach is anisotropic filtering, in which the smoothing process is based on the following governing PDE [19]:

$$\frac{\partial I_{\mathrm{m}}}{\partial t}=div(T\nabla I_m), \tag{1}$$

where I_m is the image intensity in weighting direction m, ∇ is a gradient operator, *div* is a divergence operator, t is the time. ***T*** is a *structure tensor* that provides the directionality of smoothing. It is constructed from a common *gradient tensor* ***G***, which is obtained by convolving the sum of outer products of $\nabla\ I_m$ over all weighting directions with a Gaussian kernel K_ρ:

$$\mathbf{G}=K_\rho * \sum_m(\nabla I_m \otimes \nabla I_m), \tag{2}$$

where $\otimes$ represents the outer product operator. The parameter ρ is the standard deviation of the Gaussian kernel, which determines the spatial scale of the gradient tensor.

To smooth the image isotropically inside structures and anisotropically at structure boundaries, ***T*** is defined to be a normalized inverse of ***G***. Denoting $\mathbf{v}_{\mathrm{g1}}$, $\mathbf{v}_{\mathrm{g2}}$, $\mathbf{v}_{\mathrm{g3}}$ as the eigenvectors and λ_{g1}, λ_{g2}, λ_{g3} as the corresponding eigenvalues of ***G***, the eigenvectors and eigenvalues of ***T*** are:

$$\mathbf{v}_{\mathrm{ti}}=\mathbf{v}_{\mathrm{gi}}, \quad i=1,2,3, \tag{3}$$

$$\lambda_{\mathrm{ti}}=1/\lambda_{\mathrm{gi}}, \quad i=1,2,3. \tag{4}$$

Therefore, in homogeneous regions, such as inside bundle structures, the magnitudes of the three eigenvalues of the structure tensor are comparable, yielding a similar amount of smoothing along all directions (isotropic smoothing). At the structure boundaries, the eigenvalue of the structure tensor is small across the boundaries (large intensity gradient), and large along them (small intensity gradient), thus permitting greater smoothing along the tangential direction of structure boundaries than perpendicular to it (anisotropic smoothing). To allow equal enhancement to homogeneous regions and structure boundaries, the trace of the structure tensor is normalized to be a constant *C*; consequently, the total amount of smoothing, whether isotropic or anisotropic, is the same over the entire image.

2.2. Semi-implicit scheme for anisotropic filtering

As mentioned, implementation of the anisotropic filtering typically uses an explicit scheme to solve the diffusion PDE. This however suffers from the limitation of computational inefficiency. As noted before, the computational efficiency can be boosted up by using an unconditionally stable and second order accurate Craig-Sneyd scheme for solving the PDE semi-implicitly. Before introducing this semi-implicit scheme, we first give a brief review of the explicit scheme. In order to approximate Eq. 1 numerically, we replace the time derivatives with forward-time differences and the spatial derivatives with central-space differences as follows,

$$\frac{I^{n+1}-I^{n}}{\Delta t}=\sum_{i,j=1}^{3}\partial_{x_i}(\boldsymbol{T}_{i,j}\partial_{x_j}I^{n}), \tag{5}$$

where ∂_{x_i} represents the central difference operator with respect to axis x_i, one of the spatial coordinates; I^{n} is the diffusion weighted image at time nΔt or n$^{\text{th}}$ iteration (note that the subscript *m* to *I* is suppressed to simplify notation); $\boldsymbol{T}_{i,j}$ is the (i, j) component of 3×3 structure tensor $\boldsymbol{T}$. The computational cost of each iteration is very low for the above explicit scheme, because I^{n+1} can be directly computed as follows,

$$\tilde{I}^{n+1}=\tilde{I}^{n}+\Delta t\sum_{i,j=1}^{3}L_{i,j}(\tilde{I}^{n})\tilde{I}^{n}, \tag{6}$$

where $\tilde{I}^{n}$ is a column vector representation of three dimensional image I^{n}, and $L_{i,j}$ is a linear operator matrix representing $\partial_{x_i}(\boldsymbol{T}_{i,j}\partial_{x_j})$ (see Appendix C).

In spite of the simplicity, the explicit scheme however requires a very small time step size in order to ensure its stability [24]. This translates to more iterations needed to reach a specific smoothing effect. Furthermore, as noted before, the above scheme has only first order accuracy in time.

Unlike the explicit scheme, semi-implicit schemes may not approximate time derivatives with forward-time differences. For example, a backward-time difference can also be used to replace the time derivative, and the continuous diffusion equation can be then approximated by:

$$\frac{I^{n+1}-I^{n}}{\Delta t}=\sum_{i,j=1}^{3}\partial_{x_i}(T_{i,j}\partial_{x_j}I^{n+1}), \tag{7}$$

$$(1-\Delta t\sum_{i,j=1}^{3}L_{i,j}(\tilde{I}^{n}))\tilde{I}^{n+1}=\tilde{I}^{n}. \tag{8}$$

For each iteration, a linear system must be solved to compute $\tilde{I}^{n+1}$, but the stability of the PDE is greatly improved. Note that the tensor $\boldsymbol{T}$ is computed based on $\tilde{I}^{n}$ rather than $\tilde{I}^{n+1}$, which is the reason why the scheme is called “semi-implicit” instead of “implicit”.

The semi-implicit Craig-Sneyd approach approximates Eq. 1 with the following semi-implicit scheme:

$$A\tilde{I}^{n+1}=(A+B)\tilde{I}^{n}, \tag{9}$$

where

$$A=\prod_{i=1}^{3}(1-\theta\Delta t L_{i,i}),$$
$$B=\sum_{i,j=1}^{i,j=3}\Delta t L_{i,j},$$

and θ is a real number that determines the implicitness of the scheme.

Equation 7 is still first order accurate in time because it has a mixed derivative in space and time [23]. The Craig-Sneyd scheme employs a further iteration to time-center the mixed derivative, and thus gains second order time accuracy even in the presence of the mixed derivative. The two-iteration Craig-Sneyd scheme can be summarized by

$$A\tilde{I}'^{n+1}=(A+B)\tilde{I}^{n}, \tag{10a}$$

$$A\tilde{I}^{n+1}=(A+B)\tilde{I}^{n}+\lambda M(\tilde{I}'^{n+1}-\tilde{I}^{n}), \tag{10b}$$

where

$$M=\Delta t\sum_{i,j=1(i\neq j)}^{3}L_{i,j},$$

and λ is a real number.

Equation 10a serves as an estimator that gives an approximate solution for next time step, while Eq. 10b is actually a corrector that uses a part of the estimated solution to calculate its mixed derivative. The parameter λ controls how much of the mixed derivative is computed based on the estimated solution. Such a scheme is unconditionally stable and second order accurate in both time and space for the three dimensional case when λ = ½, θ = ½ [23].

In order to solve the linear system of equations in Eq. 10a efficiently, it is split into the following three parts [23]:

$$(1-\theta\Delta t L_{1,1})\tilde{I}'^{n+1/3}=[1+(1-\theta)\Delta t L_{1,1}+\Delta t\sum_{i=2}^{3}L_{i,i}+\Delta t\sum_{i,j=1(i\neq j)}^{3}L_{i,j}]\tilde{I}^{n}, \tag{11a}$$

$$(1-\theta\Delta t L_{2,2})\tilde{I}'^{n+2/3}=\tilde{I}'^{n+1/3}-(1-\theta)\Delta t L_{2,2}\tilde{I}^{n}, \tag{11b}$$

$$(1-\theta\Delta t L_{3,3})\tilde{I}'^{n+1}=\tilde{I}'^{n+2/3}-(1-\theta)\Delta t L_{3,3}\tilde{I}^{n}, \tag{11c}$$

where $\tilde{I}'^{n+1/3}$ and $\tilde{I}'^{n+2/3}$ denote intermediate variables. All the three linear systems in Eq. 11 are composed of tridiagonal systems that can be efficiently solved by a Thomas algorithm [20].

Similarly, Eq. 10b is split into the three systems below for efficient solutions by the same Thomas algorithm:

$$(1-\theta\Delta t L_{1,1})\tilde{I}^{n+1/3}=[1+(1-\theta)\Delta t L_{1,1}+\Delta t\sum_{i=2}^{3}L_{i,i}+\Delta t(1-\lambda)\sum_{i,j=1(i\neq j)}^{3}L_{i,j}]\tilde{I}^{n}+\Delta t\lambda\sum_{i,j=1(i\neq j)}^{3}L_{i,j}\tilde{I}'^{n+1}, \tag{12a}$$

$$(1-\theta\Delta t L_{2,2})\tilde{I}^{n+2/3}=\tilde{I}^{n+1/3}-(1-\theta)\Delta t L_{2,2}\tilde{I}^{n}, \tag{12b}$$

$$(1-\theta\Delta t L_{3,3})\tilde{I}^{n+1}=\tilde{I}^{n+2/3}-(1-\theta)\Delta t L_{3,3}\tilde{I}^{n}. \tag{12c}$$

2.3. Implementation

Let σ be the noise standard deviation (SD) and Δ*t* the time step size, set *C*, the normalization constant for structure tensor, to 3, and set $\lambda = ½$, $\theta = ½$. The implementation procedures for one iteration of semi-implicit anisotropic smoothing of DWI data are summarized as follows:

1. Estimate the gradient tensor. Convolve each diffusion weighted image *I* with a zero mean, SD = σ Gaussian kernel $\boldsymbol{K}_\sigma$ to stabilize the subsequent gradient calculations, and then calculate the gradient ∇I. The gradient tensor ***G*** is estimated using Eq. 2 with $\rho = 2\sigma$.
2. Construct the structure tensor. Construct the structure tensor ***T*** using Eq. 3 and 4. As the structure tensor ***G*** is a 3×3, symmetric and positive definite matrix, its eigenvalues and eigenvectors are computed analytically (see Appendix A).
3. Implement the first iteration of the Craig-Sneyd scheme. Compute $\tilde{I}'^{n+1}$ according to Eq. 11 by using the Thomas algorithm (see Appendix B and C).
4. Implement the second iteration of the Craig-Sneyd scheme. Similar to step (3) above, $\tilde{I}^{n+1}$ is computed according to Eq. 12 also using the Thomas algorithm.

   Step (3) and (4) are performed for diffusion weighted images along each weighting direction.

2.4. Experiments with synthetic DTI data

The efficient anisotropic smoothing algorithm was first tested with a synthetic DTI dataset. The dataset consisted of two blocks of “fiber” bundles, with each block containing three identical slices of “fibers” along different orientations. Fig. 1a shows the third slice, which lies at the interface of the two blocks. The “fibers” had mean diffusivity (*D*) of 0.7×10^{-5} cm^2/s, and fractional anisotropy (FA) of 0.9, comparable to those measured *in vivo* in the human brain.

The DWIs from which diffusion tensors were computed were noise free, but were corrupted with zero mean Gaussian noise at standard deviation (SD) of 0.05, 0.1 and 0.15 times the noise free image intensity respectively. As the maximum step size for stable implementation of 3D nonlinear anisotropic filtering is 3/44 s [24], in this work 3/44 s was used as a basic time unit (dt$_0$). Anisotropic smoothing with the semi-implicit scheme was performed on the DWIs for a total simulated time of 400dt$_0$ at all noise levels with step sizes of 20dt$_0$ and 40dt$_0$. As a comparison, the same amount of smoothing with an explicit scheme was also performed on the noisy datasets with step size of dt$_0$ and 20dt$_0$.

2.5. Experiments with in vivo human DTI data

Performance of the efficient smoothing algorithm on *in vivo* data was examined with DWIs obtained from a healthy human subject. Prior to imaging, informed consent was given by the subject according to a protocol that was approved by the local ethics committee. The data were acquired with a 3T Philips Intera Achieva MR scanner (Netherlands) and an eight-element sense coil. A volume of 256×256×120 mm^3 was scanned using 32 non-collinear weighting directions and a single shot, echo-planar, pulsed gradient spin echo imaging sequence with a diffusion weighting factor (i.e., *b* value) of 1000 s/mm2. The data matrix had the size of 128×128×60, giving an isotropic resolution of 2×2×2 mm^3 in the data. To generate high SNR data used as a “gold standard” for performance evaluation, ten repeated scans were obtained, co-registered, and averaged to yield a volume dataset with an SNR of ~75. This high SNR dataset was corrupted with zero mean Gaussian noise at SD = 0.05, 0.1 and 0.15 times the noise free DWI intensity for smoothing tests. A block of seven slices in the middle of the image volume was smoothed using the semi-implicit scheme with step size of 5dt$_0$, 10dt$_0$ and 15dt$_0$ respectively, each for a total time of 60dt$_0$. As a comparison, an explicit scheme was also performed on the noisy datasets with step size of dt$_0$.

2.6. Quantitative measures of smoothing effectiveness and efficiency

Since our primary interest was the restoration of the PDD from noise corrupted images, the *effectiveness* of anisotropic smoothing was assessed based on the PDD improvement after smoothing. Defining θ_0, θ_N to be the root mean square (RMS) angular difference in the PDD with respect to noise free or “gold standard” data before smoothing and after N iterations of smoothing respectively, the PDD improvement was measured by percent RMS angular difference improvement as:

$$\frac{\theta_0-\theta_N}{\theta_0}\times 100\%. \tag{13}$$

The *Efficiency* of smoothing algorithm was evaluated in terms of total computation time required for a designated amount of anisotropic smoothing. Comparison of time efficiency between the semi-implicit and explicit schemes was made on the basis of total computation time cost for the same amount of smoothing.

3. Results

3.1. Experiments with synthetic DTI data

To demonstrate the capability of anisotropic smoothing to preserve structural boundaries while reducing noise, a region of interest (ROI) as demarked in Fig. 1a was chosen for qualitative evaluation. The ROI contains boundaries between “fiber” bundles both within the slice and across slices. Fig. 1(b–e) shows respectively an enlarged view of PDDs in the ROI in the image without noise (b), corrupted with zero mean, SD = 0.1 Gaussian noise (c), after 40 iterations of explicit anisotropic smoothing with step size dt$_0$ (d), and after one iteration of semi-implicit anisotropic smoothing with step size 40dt$_0$ (e). As can be seen, there are large variations in the orientation of the PDDs due to noise corruption; these variations are significantly reduced after one iteration of semi-implicit smoothing without appreciable blurring, i.e., mixing of the orientations of neighboring PDDs, at the boundaries. By comparison, the explicit scheme does not restore the orientation of the PPDs as faithfully as the semi-implicit scheme, albeit it has been implemented for the same total iteration time (at the expense of considerably more computation time). This is presumably attributable to its limitation of only first order accuracy.

To quantitatively assess the effectiveness of the semi-implicit smoothing scheme and compare with that of the explicit scheme, RMS angular differences in PDD and signed mean FA differences (relative to the noise free data) after various degrees of smoothing are computed for the entire slice shown in Fig. 1a. Note that the signed FA differences are used to discriminate positive and negative changes of smoothed FA with respect to true FA. Variations of the RMS angular differences and the signed mean FA differences with computation time are plotted respectively in Fig. 2 and 3 for both the explicit and semi-implicit schemes with different step sizes. For the explicit scheme, it can be observed that, after multiple iterations, the RMS angular differences in PDD are greatly reduced and the mean FA difference is stabilized around zero at all noise levels when the step size is dt_0. When the step size is $20dt_0$, both the magnitude of the RMS angular differences and FA error increase drastically, indicating that the solution to the diffusion PDE becomes unreliable and unstable. On the other hand, the semi-implicit scheme with all the step sizes removes the vast majority of the RMS angular differences after the first couple of iterations, beyond which the results are quite stable (Fig. 2). Likewise, the mean FA differences also decrease quickly and tend to stabilize around zero after the first couple of iterations (Fig. 3). These results indicate that the semi-implicit scheme is unconditionally stable for the step size up to $40dt_0$. Overall, while there are minimal residual RMS errors in the PDD, the mean FA differences tend to drop below zero, particularly for noise SD = 0.15 and step size $40dt_0$. Despite this, owing to the second order accuracy, the performance of the semi-implicit scheme with step size $40dt_0$ is still superior to that of the explicit scheme with step size dt_0 in the restoration of PDD, and its performance with step size $20dt_0$ is comparable to that of the explicit scheme with step size dt_0 in terms of FA.

In order to quantify the efficiency of the semi-implicit scheme, it was implemented for a total time of $40dt_0$ on the synthetic dataset with noise SD = 0.1 at step size $dt_{0,}$ $10dt_0$, $20dt_0$ and $40dt_{0,}$ respectively. As a comparison, the same total iteration of explicit smoothing with step size dt_0 was also performed on this dataset. The computation times on a COMPAQ Presario laptop (Mobile AMD Sempron 2800) and the PDD improvement from all these tests are summarized in Table 1. It can be seen that, due to the greater number of computations involved, the computation time for one iteration of the semi-implicit scheme is approximately eight times that of one iteration of the explicit scheme. However, the semi-implicit scheme has the advantage that much larger step sizes can be used, which translates to much fewer iterations needed for the same total amount of smoothing. This efficiency gain is clear when the computation time is compared between the semi-implicit scheme with step size $40dt_0$ and the explicit scheme with step size dt_0, which shows roughly a five fold speed-up with the semi-implicit scheme. Moreover, the table shows that the semi-implicit scheme with all step sizes is more effective in restoring the PDD than the explicit scheme for the same total iteration time, as it gives a PDD improvement of ~90% whereas the explicit scheme only gives an improvement of ~60%. Our further tests show that, even when the explicit scheme is iterated for 1000 times, which takes a total computation time of 2313 s, the PDD improvement is still below 65%. Therefore, from the point of view of the actual improvement, the efficiency gain from the semi-implicit scheme is much higher.

3.2. Experiments with in vivo human DTI data

The effect of semi-implicit anisotropic smoothing on *in vivo* data is examined in the middle (fourth) slice of the test block. The RMS angular difference in PDD and the mean FA difference relative to the "gold standard" data are computed for all voxels with FA>0.3. The voxels with low FA typically reside in the cerebrospinal fluid or gray matter, and thus have a poorly defined PDD regardless of the level of noise. Therefore, they are excluded to avoid their confounding effects on the quantitative analysis. Furthermore, as voxels near the borders of the brain may have artificially high FA values due to misregistration of images, they are removed by morphological erosion operations.

Variations of the RMS angular differences and the mean FA differences of semi-implicit smoothing with iteration time are plotted respectively in Fig. 4 and 5 for all the three noise levels and step sizes used. It can be observed that the RMS angular differences decrease after the first one or two iterations, and then begin to increase. This indicates that the effect of noise on the PDDs can only be partly eliminated by the anisotropic smoothing process. It should be noted that the remaining differences are due in part to the systematic bias in the "gold standard" data, and to the fact that the RMS angular difference in PDD itself is a biased estimator. The mean FA errors, on the other hand, change from positive to negative and tend to stabilize after the first couple of iterations. Like the angular differences, part of the negative FA errors after smoothing is attributable to the systemic bias in the original data which results in higher FA values than the ground truth. Compared to the explicit anisotropic filtering, the semi-implicit smoothing is superior in terms of best RMS PDD differences.

The gradual increase in the RMS angular differences in PDD and slowly decline in the mean FA values after the first couple of iterations are caused by excessive smoothing that smears out some of the structural details in the DTI data. Therefore, to achieve the best possible smoothing effect, an optimal number of iterations needs to be determined. This problem is in fact common to all diffusion filtering. So a major consideration in iterative smoothing is to obtain a best tradeoff between the extent of noise reduction and the level of artifacts introduced by over-smoothing (25). With this in mind, and taking the computational efficiency into consideration as well, we can readily conclude from Fig. 4 and 5 that one iteration of semi-implicit smoothing with a step size proportional to the level of image noise is optimal. For images with SD = 0.05 Gaussian noise, one iteration of smoothing with step size $5dt_0$ is most favorable in terms of both effectiveness and efficiency, whereas for images with SD = 0.1 Gaussian noise, one iteration of smoothing with step size $10dt_0$ gives the best overall performance. Using a large step size for a low level of noise would necessarily result in loss of structural details; conversely, using a small step size for a high level of noise, although yielding a slightly better smoothing effect in our case, would entail a proportionally greater number of iterations. Therefore, in applying this anisotropic filtering to the clinical DTI data, the noise level must be estimated *a priori* to determine the optimal step size.

The effect of anisotropic smoothing on the PDDs in the middle slice using the above parameter settings is illustrated in Fig. 6, in which a shows the PDDs in the original "gold standard" data, b is a enlarged view of demarked region in a, c, e and g are the PDDs corrupted with zero mean Gaussian noise at SD = 0.05, 0.1 and 0.15 respectively, and d, f and h are the corresponding restored PDDs after one step semi-implicit smoothing with step size of $5dt_0$, $10dt_0$ and $15dt_0$ respectively. It is evident that, for all the levels of noise, the disarranged orientations of the PDDs due to noise corruption are largely restored by the smoothing process. Quantitative analysis shows that the PDD improvement is 27% (1 iteration), 41% (1 iteration) and 50% (1 iteration) for noise at SD = 0.05, 0.1 and 0.15 respectively with the semi-implicit scheme. In contrast, the best PDD improvement for noise at SD = 0.05, 0.1 and 0.15 is 12% (5 iterations), 26% (11 iterations) and 37% (21 iterations) respectively for explicit anisotropic filtering.

4. Discussion and Conclusion

In this paper, we have proposed an efficient anisotropic filtering algorithm for smoothing diffusion tensor images. This efficiency is achieved by using a semi-implicit scheme for solving the diffusion filtering equation. Compared to explicit schemes, the semi-implicit scheme has unconditional stability and second order time accuracy, which allows larger step sizes to be used yet more accurate results to be obtained. This helps improve considerably the efficiency as well as enhance the effectiveness of the anisotropic smoothing algorithm. Experiments with synthetic DTI data demonstrate that the semi-implicit scheme can restore the PDD of the diffusion tensor, the parameter of major interest in this work, from noise corrupted images much more efficiently and effectively than the original explicit scheme, with a five fold speed-up and ~50% improvement in the PDD improvement. Experiments with *in vivo* DTI data also show a significant reduction of noise effects on the PDDs with a great efficiency. In particular, the *in vivo* experiments demonstrate that the best overall performance in terms of both efficiency and effectiveness can be achieved by one iteration of semi-implicit smoothing with a step size proportional to the noise level. This in effect turns the iterative smoothing algorithm into a “non-iterative” procedure.

Although in theory our algorithm is stable under any large step size, the step size of our diffusion algorithm needs to be appropriately chosen to guard against potential artifacts introduced by over-smoothing. As alluded to earlier, over-smoothing is in fact a common issue for all PDE-based image denoising methods. Technically, one can either preset the iteration time or incorporate a data fidelity term into the diffusion equation so that artifacts due to large deviations from the unsmoothed image may be prevented. The choice of iteration time and the weighting of the data fidelity term, however, need to be determined with the knowledge of noise level. Usually higher noise level entails longer iteration time and smaller fidelity weighting. In our method, we determine the iteration time similarly on the basis of the noise level, but set the step size to the optimal iteration time thanks to the unconditional stability (this results in a “non-iterative” implementation of iterative diffusion filtering). Our *in vivo* experiments suggest that 5dt0, 10dt0 and 15dt0 are the optimal step size/iteration time for images with noise level of 0.05, 0.1 and 0.15 respectively.

It should be noted that our anisotropic diffusion filtering is performed on DWIs instead of reconstructed tensor images; therefore the anticipated benefits to tissue characterization are not predicated on any particular reconstruction models (second or higher order models) to be used. A primary consideration for smoothing DWIs is that, since image noise is introduced during the image acquisition process, it is more natural and likely easier to estimate and reduce the noise on the raw DWIs. Furthermore, noise reduction on DWIs can be computationally simpler than on the images reconstructed from them, the latter of which necessarily involves additional constraints that need to be taken into special consideration.

It is worth mentioning that, in this work, construction of the structure tensor is based on analytical solutions to the eigenvalues of the gradient tensor which is positive definite and symmetric. Analytical solutions are at least five times faster than the traditional iterative solutions with negligible loss of accuracy [26]. Furthermore, solving for the eigenvalues and eigenvectors analytically allows convenient matrix operations to be performed which offers a ten fold additional speed-up with certain program languages such as Matlab.

One drawback of the semi-implicit scheme is that Eq. 10 could be split into different linear systems that generate different final outcomes. Future work could be focused on the development of a completely symmetric splitting technique to ameliorate this problem. It should also be noted that the structure tensor plays a pivotal role in anisotropic smoothing. It is constructed from the gradient tensor that is derived from image intensity gradients along the three spatial directions of the image volume. More sophisticated schemes for its estimation may be beneficial to further improvement in the effectiveness of the smoothing algorithm.

Acknowledgments

This work was supported by NIH grants RO1EB02777 (AW Anderson) and RO1EB000461 (JC Gore).

References

1. Basser PJ, Mattiello J, Le Bihan D. MR diffusion tensor spectroscopy and imaging. Biophys J 1994;66 (1):259–267. [PubMed: 8130344]
2. Le Bihan D, Mangin JF, Poupon C, Clark CA, Pappata S, Molko N, Chabriat H. Diffusion tensor imaging: concepts and applications. J Magn Reson Imag 2001;13:534–546.
3. Le Bihan D. Looking into the functional architecture of the brain with diffusion MRI. Nat Rev Neurosci 2003;4(6):469–480. [PubMed: 12778119]
4. Basser PJ, Pajevic S. Statistical artifacts in diffusion tensor MRI (DT-MRI) caused by background noise. Magn Reson Med 2000;44:41–50. [PubMed: 10893520]
5. Anderson AW. Theoretical analysis of the effects of noise on diffusion tensor imaging. Magn Reson Med 2001;46:1174–1188. [PubMed: 11746585]
6. Lazar M, Alexander AL. White matter tractography algorithms error analysis. NeuroImage 2003;20:1140–1153. [PubMed: 14568483]
7. Parker GJ, Schnabel JA, Symms MR, Werring DJ, Barker GJ. Nonlinear smoothing for reduction of systematic and random errors in diffusion tensor imaging. J Magn Reson Imag 2000;11:702–710.
8. Chen B, Hsu E. Noise removal in magnetic resonance diffusion tensor imaging. Magn Reson Med 2005;54:393–407. [PubMed: 16032670]
9. Pajevic S, Aldroubi A, Basser PJ. A continuous tensor field approximation of discrete DT-MRI data for extracting microstructural and architectural features of tissue. J Magn Reson 2002;154:85–100. [PubMed: 11820830]
10. Poupon C, Clark CA, Frouin V, Regis J, Bloch I, LeBihan D, Mangin JF. Regularization of diffusion-based direction maps for the tracking of brain white matter fascicles. NeuroImage 2000;12:184–195. [PubMed: 10913324]
11. Tench CR, Morgan PS, Blumhardt LD, Constantinescu C. Improved white matter fiber tracking using stochastic labeling. Magn Reson Med 2002;48:677–683. [PubMed: 12353285]
12. Coulon, O.; Alexander, DC.; Arridge, SA. A regularization scheme for diffusion tensor magnetic resonance images. Proceedings of 17th IPMI; 2001; p. 92-105.
13. Wang Z, Vemuri BC, Chen Y, Mareci TH. A constrained variational principle for direct estimation and smoothing of the diffusion tensor field from complex DWI. IEEE TMI 2004;23(8):930–939.
14. Tschumperle, D.; Deriche, R. Constrained and unconstrained PDEs for vector image restoration. Proceedings of 12th Scandinavian Conference on Image Analysis; 2001; p. 153-160.
15. Arsigny V, Fillard P, Pennec X, Ayache N. Log-Euclidean metrics for fast and simple calculus on diffusion tensors. Magn Reson Med 2006;56:411–421. [PubMed: 16788917]
16. Pennec X, Fillard P, Ayache N. A riemannian framework for tensor computing. Int J Comput Vision 2006;66(1):41–66.
17. Zhang, F.; Hancock, ER. Riemannian graph diffusion for DT-MRI regularization. MICCAI; 2006; p. 234-242.
18. Basu, S.; Fletcher, PT.; Whitaker, R. Rician noise removal in diffusion tensor MRI. MICCAI; 2006; p. 117-125.
19. Ding Z, Anderson AW, Gore JC. Reduction of noise in diffusion tensor images using anisotropic smoothing. Magn Reson Med 2005;49:485–490. [PubMed: 15678537]
20. Strikwerda, JC. Finite difference schemes and partial differential equations. 2. SIAM; Philadelphia: 2004.
21. Weickert J, Ter Haar Romeny BM, Viergever MA. Efficient and reliable schemes for nonlinear diffusion filtering. IEEE TIP 1998;7:398–410.
22. Barash D, Kimmel R. An accurate operator splitting scheme for nonlinear diffusion Filtering. LNCS 2106 2000:281–289.
23. Craig IJD, Sneyd AD. An alternating direction implicit scheme for parabolic systems of partial differential equations. Comput Math Appl 1990;20:53–62.
24. Gerig G, Kubler O, Kikinis R, Jolesz FA. Nonlinear anisotropic filtering of MRI data. IEEE TMI 1992;11(2):221–231.

25. Mrazek P, Navara M. Selection of optimal stopping time for nonlinear diffusion filtering. IJCV 2003;52 (2–3):189–203.

26. Hasan KM, Basser PJ, Parker DL, Alexander AL. Analytical computation of the eigenvalues and eigenvectors in DT-MRI. J Magn Reson 2001;152(1):41–47. [PubMed: 11531362]

Appendix A: Analytical solutions for eigenvalues and eigenvectors of a positive definite, symmetric matrix

Let $D=\begin{bmatrix} D_{xx}, D_{xy}, D_{xz} \\ D_{yx}, D_{yy}, D_{yz} \\ D_{zx}, D_{zy}, D_{zz} \end{bmatrix}$ denote a 3×3 positive definite, symmetric matrix. The eigenvalues and eigenvectors of *D* can be computed analytically as follows [26]:

1. Determination of the eigenvalues

The eigenvalues ($\lambda_1 > \lambda_2 > \lambda_3$) can be expressed as

$$\begin{aligned} &\lambda_1 = I_1/3 + 2\sqrt{v}\cos(\varphi) \\ &\lambda_2 = I_1/3 - 2\sqrt{v}\cos(\pi/3+\varphi), \\ &\lambda_3 = I_1/3 - 2\sqrt{v}\cos(\pi/3-\varphi) \end{aligned} \tag{14}$$

where

$$\begin{aligned} &I_1 = D_{xx} + D_{yy} + D_{zz} \\ &I_2 = (D_{xx}D_{yy} + D_{xx}D_{zz} + D_{yy}D_{zz}) - (D_{xy}^2 + D_{xz}^2 + D_{yz}^2) \\ &I_3 = D_{xx}D_{yy}D_{zz} + 2D_{xy}D_{xz}D_{yz} - (D_{zz}D_{xy}^2 + D_{yy}D_{xz}^2 + D_{xx}D_{yz}^2) \\ &v = (I_1/3)^2 - I_2/3 \\ &s = (I_1/3)^2 - I_1I_2/6 + I_3/2 \\ &\varphi = a\cos\left(\frac{s}{v}\sqrt{\frac{1}{v}}\right)/3 \end{aligned}$$

2. Determination of the eigenvectors

For i^{th} eigenvalue, the corresponding eigenvector $e_i = [e_{ix}, e_{iy}, e_{iz}]^T$ is:

$$\begin{aligned} &e_{ix} = (D_{xy}D_{yz} - B_iD_{xz})(D_{xz}D_{yz} - C_iD_{xy}) \\ &e_{iy} = (D_{xz}D_{yz} - C_iD_{xy})(D_{xz}D_{xy} - A_iD_{yz}) \\ &e_{iz} = (D_{xy}D_{yz} - B_iD_{xz})(D_{xz}D_{xy} - A_iD_{yz}) \end{aligned} \tag{15}$$

where

$$\begin{aligned} &A_i = D_{xx} - \lambda_i \\ &B_i = D_{yy} - \lambda_i. \\ &C_i = D_{zz} - \lambda_i \end{aligned}$$

To guarantee the positive-definite property, the matrix must satisfy the following conditions:

$$\begin{array}{l} I_3>0 \\ D_{ii}D_{jj}-D_{ij}^2\geq 0 \end{array} \qquad \text{for all } i,\ j=x, y,\ or\, z.$$

The eigenvalues and eigenvectors of matrices that do not satisfy the above conditions need to be calculated using iterative methods.

Appendix B: Thomas algorithm

A tridiagonal system of linear equations can be solved efficiently by Thomas Algorithm [20]. Consider the system of *m+1* linear equations with *m+1* variables (w_0, w_1…, w_m) below:

$$\begin{array}{l} a_iw_{i-1}+b_iw_i+c_iw_{i+1}=d_i, i=1, \ldots, m-1 \\ w_0=\alpha_0w_1+\beta_0 \\ w_{m-1}=\alpha_mw_m+\beta_m \end{array} \tag{16}$$

They can be reduced to

$$w_i=p_{i+1}w_{i+1}+q_{i+1}, i=0, 1, \ldots m-1. \tag{17}$$

The coefficients p_i, q_i, i = 0,1,…m can be obtained with the following code:

```
$p_0=\alpha_0, q_0=\beta_0$
$for\ i=0, 1, \ldots, m-1$
$p_{i+1}=-c_i/(a_ip_i+b_i)$
$q_{i+1}=(d_i-a_iq_i)/(a_ip_i+b_i)$
$end$
```

The variables w_0, w_1, …,w_m can then be solved with the following code:

```
$w_m=(\alpha_mq_m+\beta_m)/(1-\alpha_mp_m)$
$for\ i=m-1, m-2, \ldots, 0$
$w_i=p_{i+1}w_{i+1}+q_{i+1}$
$end$
```

Appendix C: Numerical solution for the two-iteration Craig-Sneyd scheme

Let $u_{i1,i2,i3}$ and $a^{i,\,j}{}_{i1,i2,i3}$ denote I_m (i_1h_1, i_2h_2, i_3h_3) and $T^{\,i,\,j}$ (i_1h_1, i_2h_2, i_3h_3) respectively, where h_1, h_2 and h_3 are the spatial step sizes in all the three dimensions and $i_1=0\ldots L1$, $i_2=0\ldots L2$, $i_3=0\ldots L3$.

The central difference approximation of $\partial_{x_i}(T_{i,\,j}\,\partial_{\,x_j})$ is given by

$$\partial_{x_1}(T_{1,1}\partial_{x_1})u_{i1,i2,i3}= \\ (\frac{a^{1,1}{}_{i1-1,i2,i3}+a^{1,1}{}_{i1,i2,i3}}{2h_1{}^2})u_{i1-1,i2,i3}+(-\frac{a^{1,1}{}_{i1-1,i2,i3}+2a^{1,1}{}_{i1,i2,i3}+a^{1,1}{}_{i1+1,i2,i3}}{2h_1{}^2})u_{i1,i2,i3}+(\frac{a^{1,1}{}_{i1+1,i2,i3}+a^{1,1}{}_{i1,i2,i3}}{2h_1{}^2})u_{i1+1,i2,i3}, \tag{18a}$$

$$\partial_{x_2}(T_{2,2}\partial_{x_2})u_{i1,i2,i3}= (\frac{a^{2,2}{}_{i1,i2-1,i3}+a^{2,2}{}_{i1,i2,i3}}{2h_2{}^2})u_{i1,i2-1,i3}+(-\frac{a^{2,2}{}_{i1,i2-1,i3}+2a^{2,2}{}_{i1,i2,i3}+a^{2,2}{}_{i1,i2+1,i3}}{2h_2{}^2})u_{i1,i2,i3}+(\frac{a^{2,2}{}_{i1,i2+1,i3}+a^{2,2}{}_{i1,i2,i3}}{2h_2{}^2})u_{i1,i2+1,i3}, \tag{18b}$$

$$\partial_{x_3}(T_{3,3}\partial_{x_3})u_{i1,i2,i3}= (\frac{a^{3,3}{}_{i1,i2,i3-1}+a^{3,3}{}_{i1,i2,i3}}{2h_3{}^2})u_{i1,i2,i3-1}+(-\frac{a^{3,3}{}_{i1,i2,i3-1}+2a^{3,3}{}_{i1,i2,i3}+a^{3,3}{}_{i1,i2,i3+1}}{2h_3{}^2})u_{i1,i2,i3}+(\mu_1\frac{a^{3,3}{}_{i1,i2,i3+1}+a^{3,3}{}_{i1,i2,i3}}{2h_3{}^2})u_{i1,i2,i3+1}, \tag{18c}$$

$$\partial_{x_1}(T_{1,2}\partial_{x_2})u_{i1,i2,i3}=\partial_{x_2}(T_{2,1}\partial_{x_1})u_{i1,i2,i3}= (-\frac{a^{1,2}{}_{i1-1,i2-1,i3}}{4h_1h_2})u_{i1-1,i2-1,i3}+(-\frac{a^{1,2}{}_{i1-1,i2+1,i3}}{4h_1h_2})u_{i1-1,i2+1,i3}(\frac{a^{1,2}{}_{i1+1,i2-1,i3}}{4h_1h_2})u_{i1+1,i2-1,i3}+(\frac{a^{1,2}{}_{i1+1,i2+1,i3}}{4h_1h_2})u_{i1+1,i2+1,i3}, \tag{18d}$$

$$\partial_{x_1}(T_{1,3}\partial_{x_3})u_{i1,i2,i3}=\partial_{x_3}(T_{3,1}\partial_{x_1})u_{i1,i2,i3}= (-\frac{a^{1,3}{}_{i1-1,i2,i3-1}}{4h_1h_3})u_{i1-1,i2,i3-1}+(-\frac{a^{1,3}{}_{i1-1,i2,i3+1}}{4h_1h_3})u_{i1-1,i2,i3+1}(\frac{a^{1,3}{}_{i1+1,i2,i3-1}}{4h_1h_3})u_{i1+1,i2,i3-1}+(\frac{a^{1,3}{}_{i1+1,i2,i3+1}}{4h_1h_3})u_{i1+1,i2,i3+1}, \tag{18e}$$

$$\partial_{x_2}(T_{2,3}\partial_{x_3})u_{i1,i2,i3}=\partial_{x_3}(T_{3,2}\partial_{x_2})u_{i1,i2,i3}= (-\frac{a^{2,3}{}_{i1,i2-1,i3-1}}{4h_2h_3})u_{i1,i2-1,i3-1}+(-\frac{a^{2,3}{}_{i1,i2-1,i3+1}}{4h_2h_3})u_{i1,i2-1,i3+1}(\frac{a^{2,3}{}_{i1,i2+1,i3-1}}{4h_2h_3})u_{i1,i2+1,i3-1}+(\frac{a^{2,3}{}_{i1,i2+1,i3+1}}{4h_2h_3})u_{i1,i2+1,i3+1}. \tag{18f}$$

Note that reflecting boundary condition is used when the index exceeds the dimensions of *a* or *u*. Using the above central difference approximation, the finite difference equation of Eq. 11 and 12 can then be expressed as follows,

$$(1-\theta\Delta t\partial x_1(T_{1,1}\partial x_1))u'^{n+1/3}_{i1,i2,i3}=[1+(1-\theta)\Delta t\partial x_1(T_{1,1}\partial x_1)+\Delta t\sum_{i=2}^{3}\partial x_i(T_{i,i}\partial x_i)+\Delta t\sum_{i,j=1(i\neq j)}^{3}\partial x_i(T_{i,j}\partial x_j)]u^n{}_{i1,i2,i3}, \tag{19a}$$

$$(1-\theta\Delta t\partial x_2(T_{2,2}\partial x_2))u'^{n+2/3}_{i1,i2,i3}=u'^{n+1/3}_{i1,i2,i3}-(1-\theta)\Delta t\partial x_2(T_{2,2}\partial x_2)u^n{}_{i1,i2,i3}, \tag{19b}$$

$$(1-\theta\Delta t\partial x_3(T_{3,3}\partial x_3))u'^{n+1}_{i1,i2,i3}=u'^{n+2/3}_{i1,i2,i3}-(1-\theta)\Delta t\partial x_3(T_{3,3}\partial x_3)u^n{}_{i1,i2,i3}, \tag{19c}$$

$$(1-\theta\Delta t\partial x_1(T_{1,1}\partial x_1))u^{n+1/3}_{i1,i2,i3}=[1+(1-\theta)\Delta t\partial x_1(T_{1,1}\partial x_1)+\Delta t\sum_{i=2}^{3}\partial x_i(T_{i,i}\partial x_i)+ \Delta t(1-\lambda)\sum_{i,j=1(i\neq j)}^{3}\partial x_i(T_{i,j}\partial x_j)]u^n_{i1,i2,i3}+\Delta t\lambda\sum_{i,j=1(i\neq j)}^{3}\partial x_i(T_{i,j}\partial x_j)u'^{n+1}_{i1,i2,i3}, \tag{20a}$$

$$(1-\theta\Delta t\partial x_2(T_{2,2}\partial x_2))u^{n+2/3}_{i1,i2,i3}=u^{n+1/3}_{i1,i2,i3}-(1-\theta)\Delta t\partial x_2(T_{2,2}\partial x_2)u^{n}_{i1,i2,i3}, \tag{20b}$$

$$(1-\theta\Delta t\partial x_3(T_{3,3}\partial x_3))u^{n+1}_{i1,i2,i3}=u^{n+2/3}_{i1,i2,i3}-(1-\theta)\Delta t\partial x_3(T_{3,3}\partial x_3)u^{n}_{i1,i2,i3}. \tag{20c}$$

Eq. 19a and 20a consists of (*L2+1*)×(*L3+1)* tridiagonal systems of equations, one for each value i_2 and i_3. Because those tridiagonal systems exhibit exactly the same form as Eq. 17, the Thomas algorithm can be used to solve the values of $u_{i1,\ i2,\ i3}$ for each value i_2 and i_3, Usually the Thomas algorithm is implemented in a two-loop on i_2 and i_3. Similarly, Eq. 19b and 20b consists of (*L1+1*)×(*L3+1*) tridiagonal systems of equations, and Eq. 19c and 20c consists of (*L1+1)×(L2+1)* tridiagonal systems of equations. Those tridiagonal systems can be solved by the Thomas algorithm in a similar way.

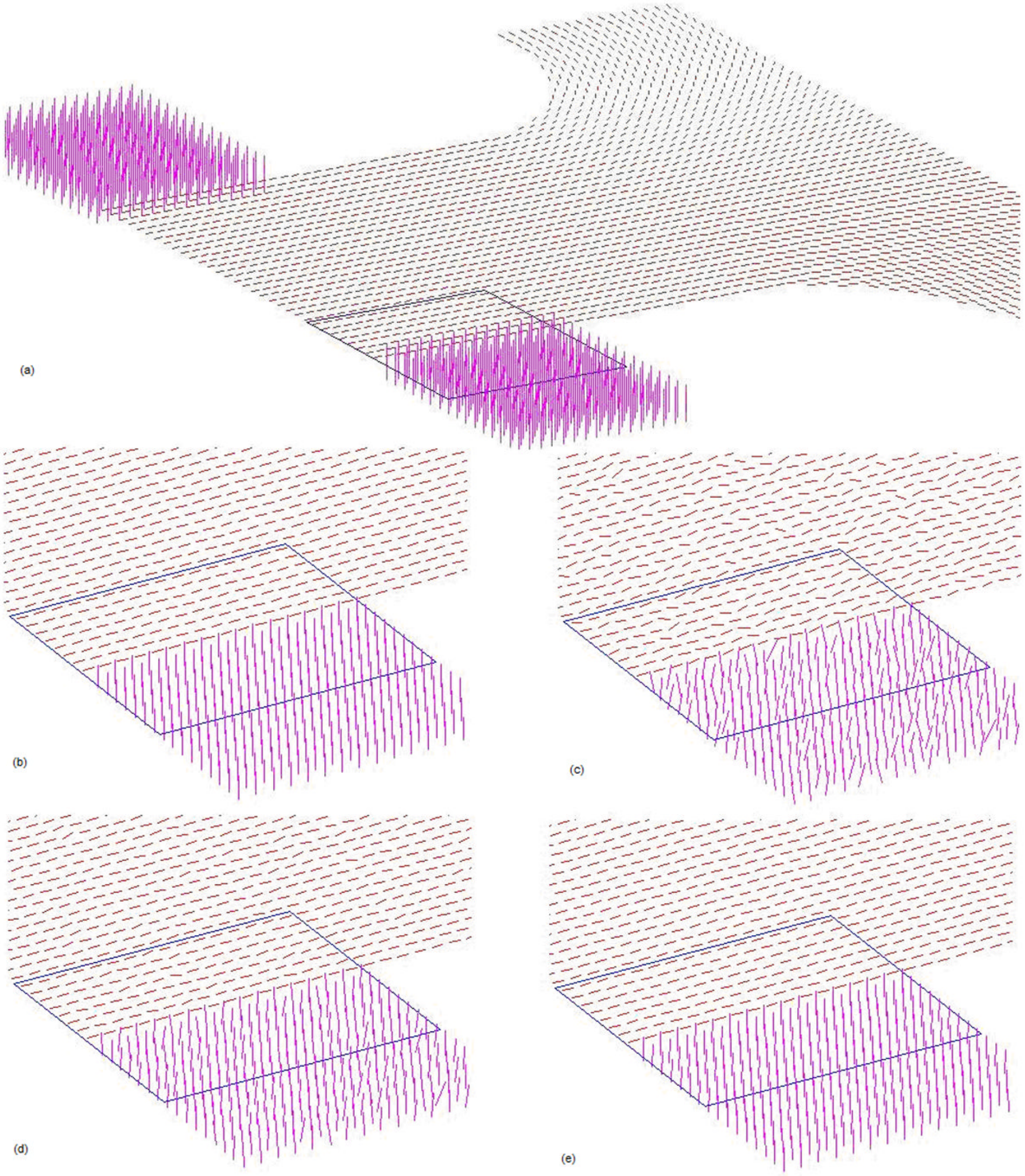

Figure 1.

Illustration of the effect of anisotropic smoothing on synthetic data: (a) a middle (third) slice of the original noise free data. The line segments represent the tensor PDD; (b) enlarged view of the PDDs in an ROI demarked in (a); (c) PDDs in the ROI corrupted with zero mean, SD = 0.1 Gaussian noise. (d) PDDs after 40 iterations of explicit anisotropic smoothing with step size dt_0; (e) PDDs after one iteration of semi-implicit anisotropic smoothing with step size $40dt_0$.

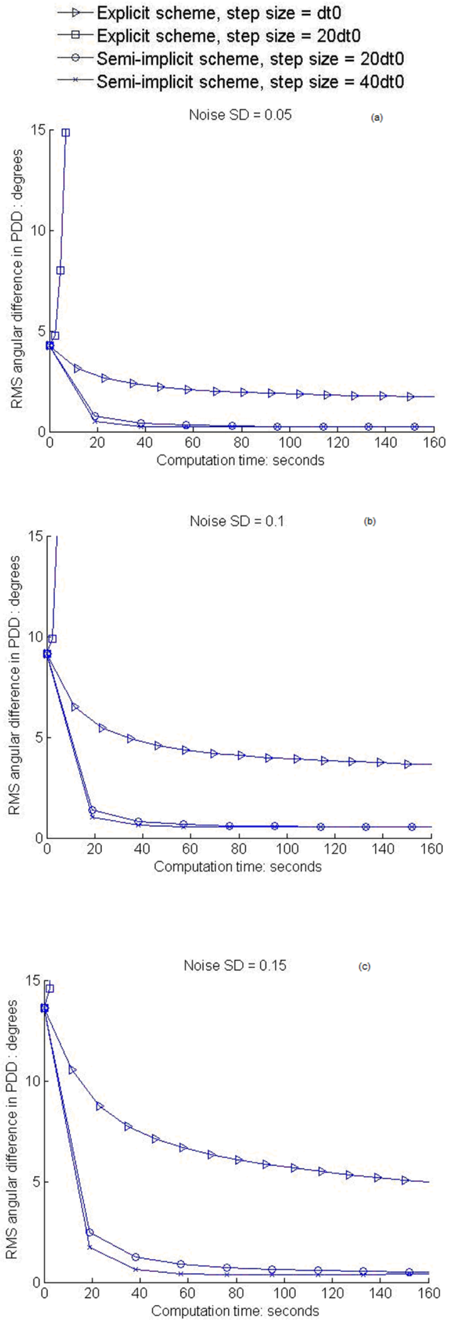

Figure 2.

Variations of RMS angular difference in PDD with computation time of smoothing. (a–c) show the results for the synthetic data at the noise level SD = 0.05, 0.1, 0.15 respectively. The portions of the curves with PDD difference above 15 degrees are not shown.

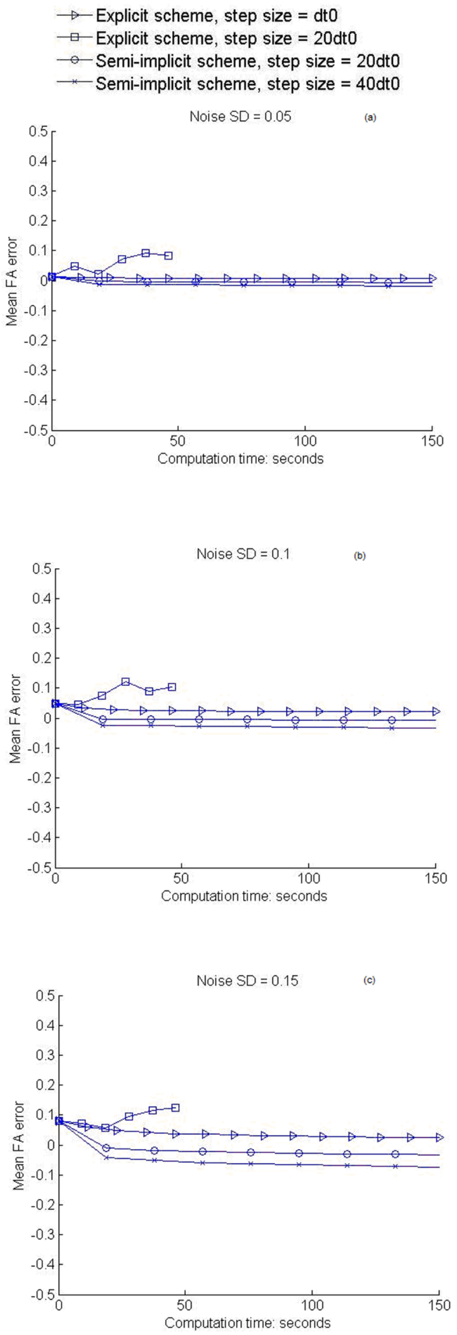

Figure 3.

Variations of mean FA difference with computation time of smoothing. (a–c) show the results for the synthetic data at the noise level SD = 0.05, 0.1, 0.15 respectively. The curve of explicit scheme with step size 20dt$_0$ is shorter than other curves because of the small number of iterations used and the low computation time of each iteration.

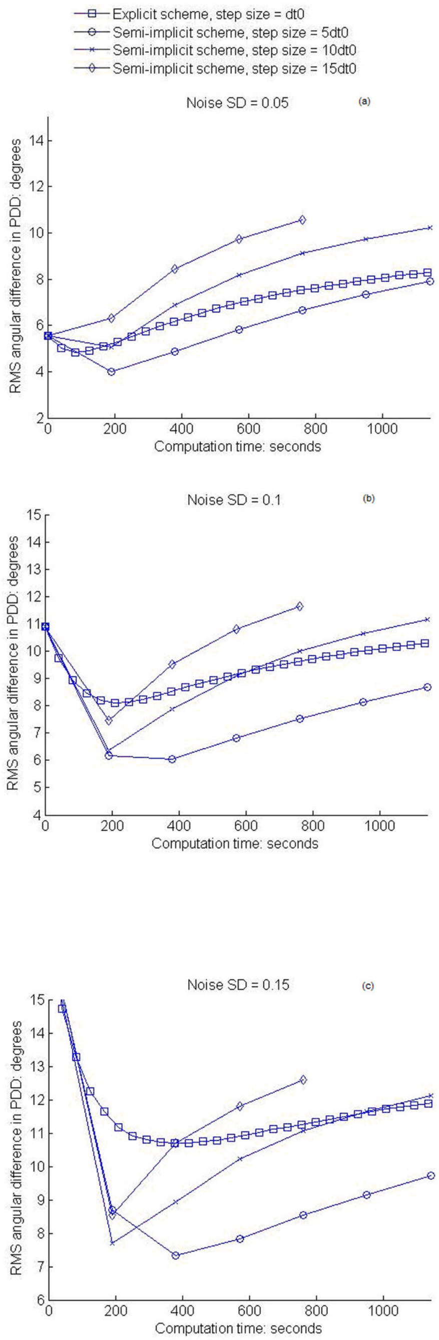

Figure 4.

Variations of RMS angular difference in PDD with computation time of smoothing. (a–c) show the results for the *in vivo* human data at the noise level SD = 0.05, 0.1, 0.15 respectively.

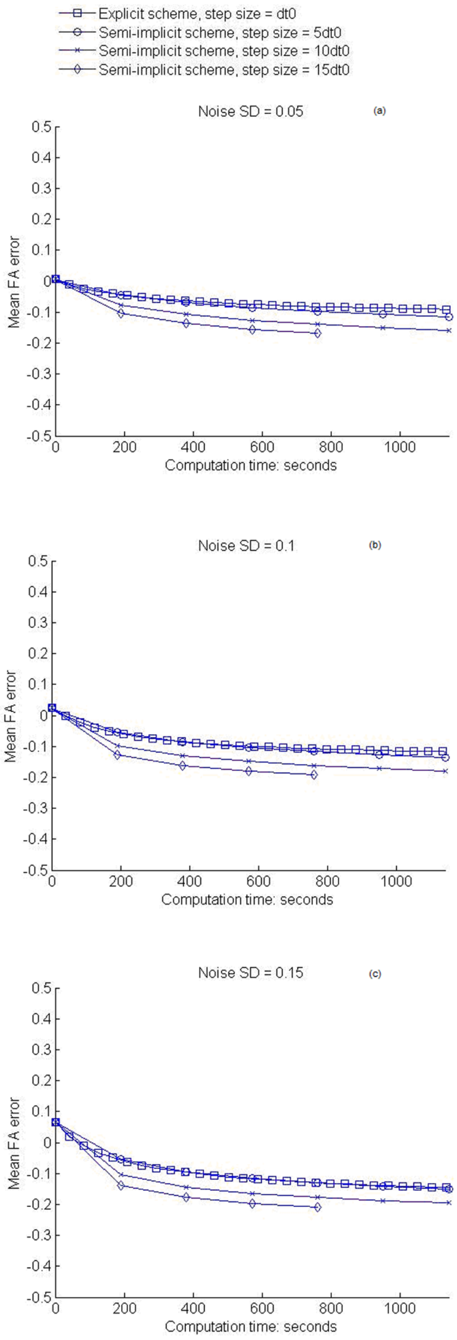

Figure 5.

Variations of mean FA difference with computation time of smoothing. (a–c) show the results for the *in vivo* human data at the noise level SD = 0.05, 0.1, 0.15 respectively.

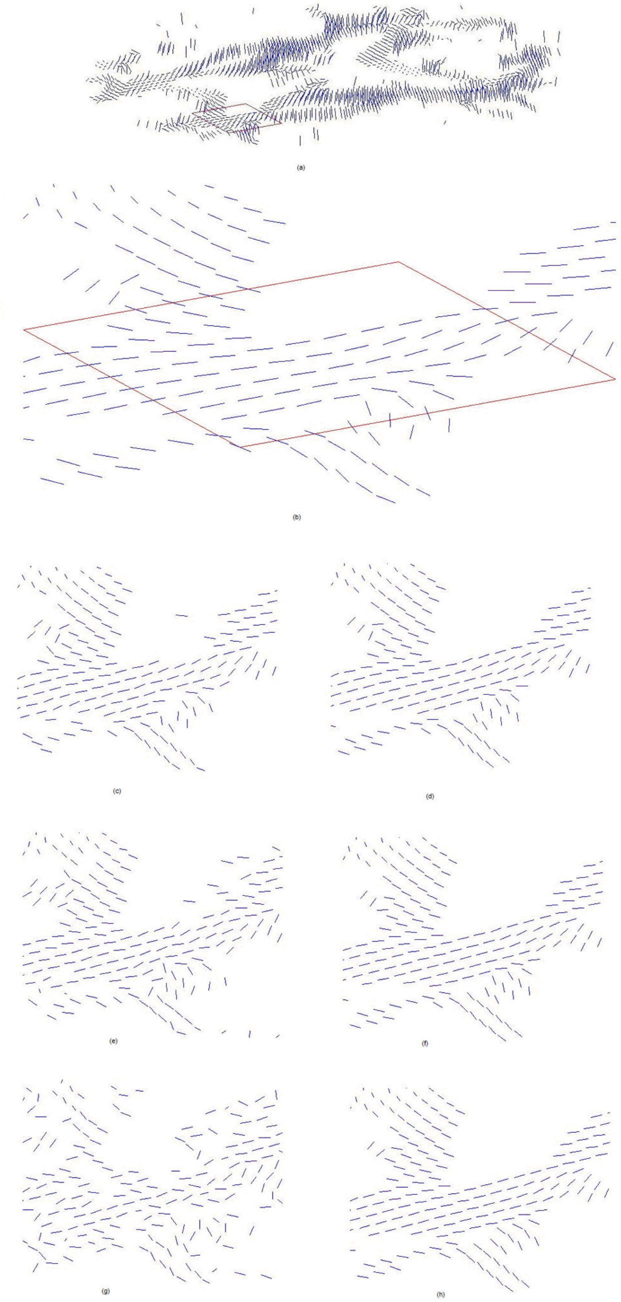

Figure 6.

Illustration of the effect of semi-implicit anisotropic smoothing on *in vivo* human data: (a) the middle (fourth) slice from the original high SNR data; (b) enlarged view of the PDDs in an ROI demarked in (a); (c) PDDs corrupted with zero mean, SD = 0.05 Gaussian noise; (d) PDDs after one iteration of smoothing on (c) with step size $5dt_0$; (e) PDDs corrupted with zero mean, SD = 0.1 Gaussian noise; (f) PDDs after one iteration of smoothing on (e) with step size $10dt_0$; (g) PDDs corrupted with zero mean, SD = 0.15 Gaussian noise; (h) PDDs after one iteration of smoothing on (g) with step size $15dt_0$.

Table 1

Comparisons of computation time and percent angular difference improvement of the PDD between the semi-implicit and explicit schemes

Semi-implicit scheme, total time = 40dt$_0$	**Computation time**	**Percent angular difference**
Step size = dt$_0$	768 s	90.6%
Step size = 10dt$_0$	76 s	92.3%
Step size = 20dt$_0$	33 s	93.3%
Step size = 40dt$_0$	19 s	93.0%
Explicit scheme, total time = 40dt$_0$		
Step size = dt$_0$	92 s	61.0%