A NEW PROPOSAL FOR 3D FIBER TRACKING IN SYNTHETIC DIFFUSION TENSOR MAGNETIC RESONANCE IMAGES

L. M. San-José-Revuelta, M. Martín-Fernández, C. Alberola-López
Dept. Teoría de la Señal y Comunicaciones e I.T., University of Valladolid, SPAIN
Phone: +34 983 423660, Fax: +34 983 423667
{l sanjose, marcma, caralb}@tel.uva.es

ABSTRACT
In this paper a new fiber tracking algorithm to be used with diffusion tensor fields acquired via magnetic resonance imaging is developed. The research effort in tracking fibers of nervous cells in the human brain affords many benefits to the medical community. The proposed method is an improvement over a previously developed algorithm that uses both geometrical and probabilistic criteria. The new scheme offers better results in regions with branching and crossing fibers, showing better computational and robustness properties. Like its predecessor, it also offers the capability of reporting an uncertainty value for the computed tracts.

1. INTRODUCTION
The recent development of Magnetic Resonance Imaging (MRI) has led to the design of numerous imaging techniques. One of these is Diffusion Tensor MRI (DT-MRI), which measures the diffusion of hydrogen atoms within water molecules in the 3D space. Since in cerebral white matter most random motion of water molecules are restricted by axonal membranes and myelin sheets, diffusion anisotropy allows depiction of directional anisotropy within neural fiber structures [1]. DT-MRI has raised great interest in the neuro-science community for a better understanding of the fiber tract anatomy of the human brain as well as the connectivity between different parts of the brain.

Apart from a very few approaches for direct volume rendering [2], the great majority of DTI visualization techniques focuses on the integration of sample points along fiber trajectories and their three-dimensional representation [3]. These approaches (called fiber tracking) usually make use only of the principal eigenvector of the diffusion ellipsoid as an estimate of the predominant direction of water diffusion in a voxel [1]. Several approaches have already been developed. A brief review of related work can be found in [4]. Some of them include the Runge-Kutta approach, the multiple diffusion tensor approach, the tensorline approach and the exhaustive search approach, to name a few.

The in vivo visualization of fiber tracts opens up new perspectives for neurological research. Nevertheless and due to some deficiencies in these tracking algorithms and several shortcomings inherent in datasets, they may depict fiber tracts which do not exist in reality or miss to visualize important connectivity features. Thus, the user of the system (viewer) must be provided with the uncertainty of the presence of a depicted fiber in a certain location. This paper continues upon a previous tracking algorithm [4] which takes into account the complete 3D information provided by the diffusion matrix about the likelihood of continuing the path through every possible future direction (voxel hypotheses) instead of only considering the principal diffusion direction (PDD).

The novelty in this paper consists in the introduction of a new procedure that adapts the number of offspring paths emerging from the actual voxel $V_i$, to the degree of anisotropy observed in the proximity of $V_i$. This will improve the estimation robustness in areas where multiple fibers cross and it will also keep complexity to a moderate level.

2. ALGORITHM DESCRIPTION
We propose a tracking algorithm that uses both geometrical and probabilistic criteria and iterates over several points in the analyzed volume. Specifically, these points are voxels given by the highest probabilities in the previous iteration. The algorithm starts in a seed voxel, $V_0$, which is specified by the user. At every iteration, the method evaluates a set of parameters related to the central voxel of a cubic structure consisting of $3 \times 3 \times 3 = 27$ voxels. The central point, $V_c$, represents the last point of the tract being analyzed. In the first iteration $V_1 = V_0$. There exist 26 possible directions to take for the next iteration (in order to select the next point of the tract). Once a list of valid points is available (which excludes the previous point and those points out of the volume), a measure $P_v$, $I \in \{\text{valid points}\}$, is evaluated for each of them based on a probability value of going from voxel $V_c$ to voxel $V_i$. This probability takes into account the eigenvalues and eigenvectors available at point $V_c$ from the DT-MR image diffusion matrix. Fig. 1 is used in order to calculate this probability. The table on the right shows, for each of the 26 voxels, the changes in indexes $(m, n, p)$, when a tract goes from voxel $V_c$ to voxel $V_i$. For instance: when going from point No. 14 ($V_0$) to No. 6, coordinate $m$ reduces by 1, $n$ remains the same, and $p$ increases by 1. This is represented in Fig. 1, right with $\pi_m \pi_n \pi_p = (-0+)$. With the information obtained from Fig. 1 the probability of each possible destination $V_i$ can be calculated taking into account the projection of each of the eigenvectors to each of the directions defined in the triplet $\pi_m \pi_n \pi_p$. Besides, each projection is weighted by the corresponding eigenvalue $\lambda$. Thus, in the previous example, $P_i$
Fig. 1. Modifications of indexes (m, n, p) when moving from \( V_i \) to the neighboring voxel \( V_j \), \( 1 \leq i \leq 27, \ i \neq 14 \).

should be calculated as: \( P_i = V_{i1} \hat{\lambda}_1 + V_{i2} \hat{\lambda}_2 + V_{i3} \hat{\lambda}_3 + V_{i1} \hat{\lambda}_4 + V_{i2} \hat{\lambda}_5 + V_{i3} \hat{\lambda}_6 \), where \( V_{ik} \) represents the \( x \)-component of eigenvector \( j, 1 \leq j \leq 3 \) (assuming that \( \hat{\lambda}_1 > \hat{\lambda}_2 > \hat{\lambda}_3 \)), and so on. In the general case

\[
P_i = \sum_{x,y,z} \sum_{j=1}^{8} V_{ij} \lambda_j
\]

with \( \chi_v, \chi_c, \chi_s \) being zero if \( \pi_m, \pi_n, \pi_p \) are zero, respectively, and equal to 1 otherwise.

The axes reference criterion for the \( (x, y, z) \) vector components is also shown in Fig. 1, left. Note that, for this calculus, the sign “−” in the triplet is equivalent to sign “+”. In order to properly weight and calculate Eq. (1), \( P_i \) must be weighted by 0.33 if no zeros are present in triplet \( i \), and by 0.5 if there is one zero.

2.1. Local and path probabilities

The following anisotropy index is used in the algorithm

\[
fa = \sqrt{\frac{(\hat{\lambda}_1 - \hat{\lambda}_2)^2 + (\hat{\lambda}_2 - \hat{\lambda}_3)^2 + (\hat{\lambda}_3 - \hat{\lambda}_4)^2}{2(\hat{\lambda}_1^2 + \hat{\lambda}_2^2 + \hat{\lambda}_3^2)}}
\]

with \( \hat{\lambda}_1 \geq \hat{\lambda}_2 \geq \hat{\lambda}_3 \). If both \( fa(V_i) \) and \( fa(V) \) do not exceed a certain threshold (0.2), then point \( V_i \) is eliminated as a possible destination point.

Taking into account both the probability given by Eq. (1) and the anisotropy given by Eq. (2), we define the local probability of voxel \( i, i \in \{ \text{valid points} \} \), as

\[
P_i^1 = a \cdot \mu_1 \cdot fa(V_i) + (1-a) \cdot \mu_2 \cdot P_i
\]

where parameter \( a \) allows the user to give a higher relative weight to either the anisotropy or to the local probability, and \( \mu_1 \) and \( \mu_2 \) are scaling factors (normally, 1 and 1000, respectively).

The probability \( P_i^1 \) of each voxel can be recursively accumulated, yielding the probability of the path generated by the successive values of \( V_c \),

\[
P_i^m(k) = P_i^m \cdot P_i(k-1)
\]

with \( k \) being the iteration number, and \( P_i^m \) the value of \( P_i^1 \) normalized to its highest value.

2.2. Eigenvectors and direction considerations

The final probability of voxel \( i \) makes also use of the so-called smoothness parameters (described in [5]) which judge the coherence of fiber directions among the trajectories passing through voxel \( V_c \),

\[
sp_i = \hat{\psi}(V_c(k)) \cdot \hat{\psi}(V_c(k-1))
\]

2.3. Pool of surviving voxels (next seeds)

Taking into account both criteria given by Eqs. (4) and (9), the follow metric is calculated and used for selecting the best \( s \) voxels

\[
P(i) = 0.5 \left( \frac{P_c(i)}{\max \ P_c(i)} + \frac{P_s(i)}{\max \ P_S(i)} \right)
\]

A pool of voxels is formed by selecting, at the end of each iteration, the \( s \) best voxels according to Eq. (10). The first voxel of the pool becomes the central voxel \( V_c \) at the beginning of the next iteration (afterwards, it is removed from the pool) expanding this way the current pathway.
At this point of the algorithm is where the main novelty of this paper is introduced. Instead of using a fixed number of surviving voxels, \( s \)—this was the strategy used in [4]—now we propose to set the value of \( s \) depending on the degree of anisotropy found in \( V_c \) and its surroundings. When this anisotropy is high, it means that a high directivity is present in that zone, and the probability that \( V_c \) belongs to a region where fibers cross is really low. Thus, \( s \) must take a small value (1, 2 or 3). On the other hand, if \( V_c \) is found to be situated in a region of low anisotropy, the probability of having fibers crossing or branching is higher. In this case, it is interesting to explore various paths starting in \( V_c \). This can be achieved by setting parameter \( s \) to a higher value (4 to 10).

This idea is illustrated in Fig. 3, where point \( V_0 \) belongs to a very anisotropic region. In this case, \( \mathbf{v}(V(k-1)) \) and \( \mathbf{v}(V(k)) \) are nearly parallel and parameter \( s_p \) is close to 1. The pool of surviving voxels should be augmented with those points marked with a white filled triangle. On the other hand, points like \( V'_c \), which are situated in a region where two orthogonal fiber bundles cross, has a small anisotropy. In this case, all the points marked with black triangles could be added to the pool so that they will be considered as seeds in future iterations. Numerical simulations show that the two main consequences of the on-line adjustment of \( s \) are: (i) a better use of computational resources, and (ii) a better performance in regions with crossing or branching fibers.

3. NUMERICAL RESULTS

In order to assess the properties of the tracking algorithm, we have generated three different synthetic DT-MRI data in a 50 × 50 × 50 grid, where the true path of the fiber tract is known.

To make the simulated field more realistic, Rician noise was added in the diffusion weighted images which were calculated by a direct application of the Stejskal-Tanner equation [6]. The three test images considered are shown in Fig. 4. The desired noisy synthetic diffusion tensor data was obtained using an analytic solution for the inverse Stejskal-Tanner equation [6]. The eigenvectors in the isotropic areas were \( \lambda_1 = \lambda_2 = \lambda_3 \), while in the remaining voxels of the image \( \lambda_1 = 7, \lambda_2 = 2 \) and \( \lambda_3 = 1 \). Fig. 5(a) shows the tracing results for the “cross” test image, for different selections of the initial seed \( V_0 \). The two parts of the cross are located as follows: one of them is parallel to the \( x \) axis, with \( 20 \leq y, z \leq 30 \), and the other is parallel to the \( y \) axis, with \( 20 \leq x, z \leq 30 \). It can be seen how the algorithm can be designed in order to pass through isotropic zones where different fiber bundles cross \( 20 \leq x, y, z \leq 30 \). It is also appreciated how the algorithm differentiates between the totally isotropic zones extrinsic to the tracts and the fiber bundles.

The differentiation between voxels belonging to a fiber or to a very isotropic area, respectively, is attained by mapping the path probabilities given by Eq. (4) into a color scale and classifying them according to some fixed thresholds. Three different seeds \( (S_1, S_2, S_3) \) are shown in Fig. 5(a). \( S_1 \) and \( S_2 \) belong to the intrinsic volume (voxels with a very high anisotropy) and the algorithm moves through the most probable direction following the main direction of the cross in each situation. On the other hand, when an extrinsic point such as \( S_3 \) is selected as seed, the algorithm explores in the neighboring voxels until it finds a voxel with a high anisotropy value (point \( P_1 \)). Once \( P_1 \) is found, the tracking algorithm proceeds as in the case of \( S_1 \) and \( S_2 \).

Results for the “earth” and “log” images are shown in Fig. 5(b)–(c), respectively. In the first case, the fibers look like geographical parallels of different spheres with the same center and different radius, situated on plane \( x-y \) and with a totally isotropic cylindrical volume in the center, parallel to the \( z \) axis (Fig. 4, center), whereas in the “log” case the volumetric image consists of logarithmic curves symmetrically distributed around the \( y \) axis (this scenario constitutes the most complicated situation since the diffusion field experiments variations with the three coordinate axes). Fig. 5 shows that, in both cases, the algorithm finds again the proper fiber path whatever (extrinsic or intrinsic) seed voxel is chosen. Notice that, the extrinsic seeds \( S_i \) are located far away from the fiber bundles region, thus making the algorithm explore a wider range of points before reaching the points \( P_i \) that belong to an existing fiber path.

Finally, we study and compare the robustness of the tracking algorithm when: (i) parameter \( s \) is fixed during the whole estimation of the path, and (ii) parameter \( s \) is dynamically changed depending on the anisotropy:
Parameter $s$ fixed. These simulations were carried out with $s = 2$. The plots depicted in Fig. 5 correspond to this situation. Notice that, in the case of the cross-like image (Fig. 5, left) the algorithm starts in $S_1$, $S_2$ or $S_3$ (in this later case it first reaches point $P_1$) and it follows a path which passes straight ahead through the crossing volume ($20 \leq x, y \leq 30$) without considering a branching situation. The results would be almost the same considering $s = 1$. The convergence performance for different SNR values are shown in Table 1. It can be seen how the algorithm converges properly within a very wide range of SNRs. The percentages obtained for the “cross” and the “earth” test images are very close, while for the “log” case the algorithm exhibits a slightly lower convergence.

Parameter $s$ depends on the anisotropy. Fig. 6, left shows the results for the “cross” image when $s = 6$ for high anisotropy and $s = 2$ otherwise. The bundle is estimated and branching is properly managed. Setting the smoothness parameters more restrictively, a single fiber bundle can be estimated, as shown in Fig. 6, right.

Consequently, the robustness of the algorithm in branching and crossing situations are now more flexibly controlled and the computational load can be maintained to its lowest value for the kind of desired estimation.

4. CONCLUSIONS

An algorithm for fiber tracking has been developed and tested with several synthetic DT-MRI data. This algorithm combines both the probability of advancing in a specific direction based on the projection of all the eigenvectors components into the corresponding directions (making use of more information than merely the principal eigenvector) and four smoothness criteria based on the relation between the voxels and eigenvectors’ orientations. The number of possible seeds considered for next iterations (possible path emerging from the actual point) are on-line adjusted based on the anisotropy properties in the local context. Simulations showed that this modification leads to more robust estimations in branching and crossing situations.

5. ACKNOWLEDGEMENTS

The authors acknowledge the Spanish CICYT for project TEC2004-06647-C03-01, and the European Commission for the Network of Excellence SIMILAR (FP6-507609).

6. REFERENCES


