DIFFUSION TENSOR FIELD VISUALIZATION

by

Sıla Girgin

B.S., Computer Engineering, Boğaziçi University, 2005

Submitted to the Institute for Graduate Studies in
Science and Engineering in partial fulfillment of
the requirements for the degree of
Master of Science

Graduate Program in Computer Engineering

Boğaziçi University

2008

ACKNOWLEDGEMENTS

I would like to thank to my thesis supervisor Dr. Ali Vahit Şahiner for his endless support, encouragement and guidance. I would like to thank Assist. Prof. Burak Acar for his endless support, suggestions and revisions on the thesis. I feel fortunate of being one of their research students throughout my study. I would like to thank Prof. Lale Akarun for participating in my thesis committee.

This thesis has been supported in part by TUBITAK KARIYER-DRESS (104E035) project and Bogazici University B.A.P. (07A203) project grants.

I want to thank VAVLab members - Erkin Tekeli and Enver Yağcı- for their friendship and encouragement.

I am very grateful to my family and my special friends for their endless love and support.

ABSTRACT

DIFFUSION TENSOR FIELD VISUALIZATION

Visualization of diffusion in restricted media, such as the fiber network in human brain, via MR imaging provides a better understanding of the underlying micro structure and allows us to study it in detail. However, 3D tensor field visualization presents several challenges.

In this study, we present an interactive tool for visualizing 3D diffusion tensor fields. The tool is composed of two approaches. The first one, the Tensor Paint, is based on the definition of local connectivity between neighboring tensor pairs. The connectivity value together with a threshold determines whether a tensor passes the paint it has received to its neighbors or not. Tensor painting employs a number of different connectivity definitions. They are all based on the interpretation of the diffusion tensors as the covariance matrices of the 3D Gaussian PDFs representing the spatial distribution of moving particles. The second approach is based on the modulation of an input white noise texture by the tensor field. For this purpose, we have utilized the Line Integral Convolution (LIC) technique which is a widely used technique in flow visualization. In LIC, the application of a convolution filter follows local streamline computation.

In the study, we first discuss the requirements for the visualization of diffusion tensor fields and then present our tool. Presentation of the results and evaluations obtained on two phantom diffusion tensor fields and a discussion follows this.

ÖZET

DİFÜZYON TENSÖR ALANI GÖRSELLEŞTİRME

MR resimleme vasıtasıyla difuzyonun sınırlandırıldığı alanlarda, mesela insan beyninin içindeki sinir ağlarında, göreselleştirilmesi alttaki mikro yapıyı anlamamızı sağlar ve onu daha detaylı incelememize izin verir. Ancak 3 boyutta tensör alanı görselleştirme bir çok meydan okumayı içinde barındırır.

Bu calışmada, 3 boyutlu difuzyon tensör alanı görselleştirmek için interaktif bir uygulama sunulmuştur. Uygulama iki yaklaşım içermektedir. Birincisi, Tensör Boyama, komşu tensor çiftlerinin lokal bağlılık tanımlarına dayanmaktadır. Lokal bağlılık değerleri, eşik değerleri ile birlikte bir tensörün kendisine ulaşan boyayı komşu tensörlere ulaştırıp ulaştırmayacağını belirler. Tensör boyama birçok değişik bağlılık tanımını kullanır. Bu tanımların hepsi, difuzyon tensörünün, hareket eden parçacıkların uzaysal dağılımını açıklayan 3 boyutlu Gaussian PDF lerin ortak değişinti matrisi olarak yorumlanmasına dayanır. Ikinci yaklaşım tensör alanına uygun olarak beyaz gürültü imgesinin düzenlenmesine dayanır. Bunun için akış görselleştirmede çok sık kullanılan Line Integral Convolution(LIC) tekniğinden faydalanılmıştır. LIC metodunda, lokal akış çizgilerinin hesaplanmasının ardından evrişim filtresi kullanılır.

Bu çalışmada, önce difuzyon tensörünün görselleştirilmesinin gereklilikleri ele alındı ve ardından geliştirilen uygulama sunuldu. İki phantom difuzyon tensör alanı üzerinden elde edilen sonuçların ve değerlendirmelerinin sunumu ve yorumlama kısmı bunları takip etti.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF SYMBOLS/ABBREVIATIONS

c_l	linear anisotropy coefficient of diffusion tensor
c_p	planar anisotropy coefficient of diffusion tensor
c_s	spherical anisotropy coefficient of diffusion tensor
e_l	first eigenvector
e_2	second eigenvector
e_3	third eigenvector
λ_l	first eigenvalue
λ_2	second eigenvalue
λ_3	third eigenvalue

DTI	Diffusion Tensor Image
FA	Fractional Anisotropy
LIC	Line Integral Convolution
MR	Magnetic Resonance
PDD	Principle Diffusion Direction
PDF	Probability Distribution Function
SNR	Signal to oise Ratio

1. INTRODUCTION

The improvements in data acquisition technologies in science, medicine and engineering has generated the need to deal with large amounts of data effectively and efficiently. Analyzing, understanding and utilizing huge datasets is one of the main challenges in all application areas. Visualization is one of the most important instruments that can be utilized for this purpose and it is essential to understand mathematical models of complex phenomena. Visualization reduces and refines data rapidly and economically, enabling us to eliminate large volumes of redundant data. Advances in visualization has enabled researchers to analyze and understand experimental, simulated, and observational data, and through this understanding, to address problems previously thought to be unsolvable.

Visualization provides a quick and better understanding of the quantitative information represented by the data[1]. Since 1750's, visualization in the form of data graphics has been widely used to visually display measured quantities[34]. Today, improvements in computer graphics and hardware technology, allow us to visualize large amounts of datasets interactively. We can use visualization tools to detect patterns, assess situations and prioritize tasks.

The subject of this thesis is visualization of diffusion tensor fields. A diffusion tensor field is a mathematical entity that represents the motion of water molecules in a 3D environment. Water molecules act differently in restricted and unrestricted media. Their behavior in restricted media is a function of the structure. Thus by visualizing the diffusion tensor field, we can analyze and have a better understanding of this structure. Main difficulty in visualization of a diffusion tensor field is due to its 3D and multi-variate nature.

Diffusion Tensor Magnetic Resonance Imaging (DT-MRI or DTI) is a medical application area which utilizes diffusion tensor field data. More specifically, a DT-MRI image reflects the internal structure of brain that is the location and orientation of

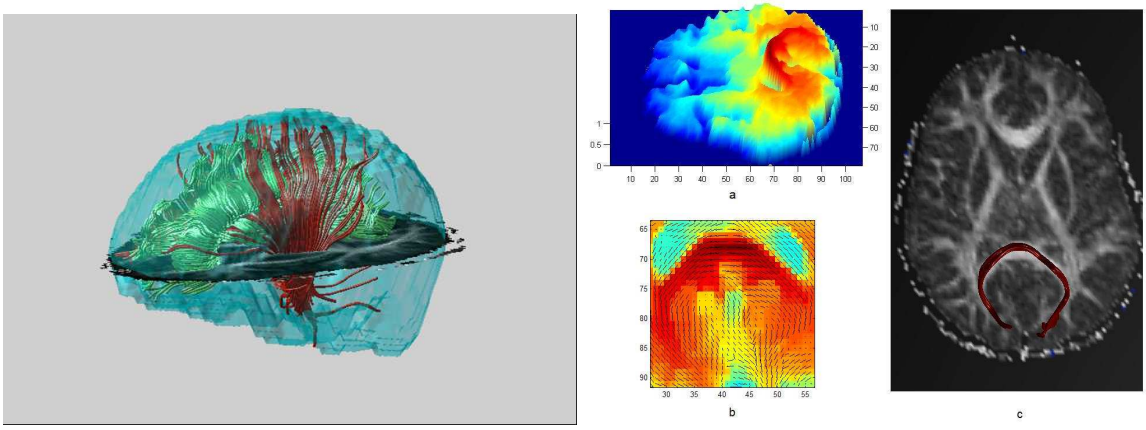

Figure 1.1. DTI Visualization Examples[3]

nerve fibers. Today, it is widely used for diagnosis purposes.

Diffusion tensor fields are usually represented by using scalar metrics. These metrics are parameters that characterize the diffusion information, and they can be calculated from the tensor. A diffusion tensor is represented by six parameters in three dimension. 3D geometric objects called glyphs are used for visualizing tensors.[6] Parameters that define a tensor can be visualized by modulating the color, shape, and orientation of a glyph. In addition to this, tensor fields can also be visualized on the basis of their topology. The diffusion through the field can globally be visualized by lines and surfaces[11]. Texture modification and volume rendering are other examples of visualization techniques employed[14, 15].

In this thesis, for visualizing diffusion tensor fields, we have employed a set of well-known methods as well as two newly proposed methods. To test the methods and to analyze the results, a graphical tool is developed. The tool is implemented as an Qt-application using C++[30, 32]. For 3D visualization, OpenGL is used[31]. A phantom diffusion tensor field is used in algorithm development. Firstly, basic visualization methods are used. Then two new methods that are based on the concepts of connectivity and flow paths are proposed. Evaluations are reported for each method and data set. In addition to the phantom data, a set of MR-DTI data(diffusion tensor

field for brain) is also visualized by using the same methods.

The thesis is organized as follows: literature review on diffusion tensor field visualization is given in Chapter 2. In Chapter 3, two main concepts, connectivity and flow paths, are discussed and the visualization methods are described in detail. Chapter 4 gives the evaluations carried out on the phantom data and result obtained. Finally, the last chapter concludes the study with a discussion of the performed work and the future research studies.

2. TENSOR VISUALIZATION

2.1. Background

2.1.1. Tensor

A scalar field describes a one-to-one correspondence between a single scalar number and a point in the field. An n-dimensional vector field is described by a one-to-one correspondence between n-numbers and a point in the field. Tensor field is a generalization where n-squared number or n-cubed number is assigned to a single point(a tensor) in the field. More precisely, a tensor field is a function that assigns a tensor to every geometric space point.[2]

A ρ order tensor is a collection of n^ρ numbers or functions which characterize a certain property of point in n-dimensional space. In general, scalar fields are referred to as tensor fields of order zero whereas vector fields are called tensor fields of order one.

Tensor fields, especially second-order tensor fields, are useful in many medical, mechanical and physical applications. Effective tensor visualization methods can enhance research in a wide variety of fields. However developing an effective algorithm can be difficult because of the variety of information contained in 3D tensor fields.

Tensor fields subject to this thesis, are diffusion tensor fiels which symmetric positive semi-definite second order tensor fields. These tensors which characterize diffusion in 3D can be represented as symmetric semi-definite 3×3 matrices. To understand what diffusion tensor represents physically, we should explain the meaning of diffusion and how it is represented mathematically by using a tensor.

2.1.2. Diffusion

In short Fick's Law says that there is a flux of concentration in the negative concentration gradient direction proportional to a constant coefficient (Diffusion Coefficient) which is set by the environment.

According to Fick's law, flux F is:

$$F = -D\nabla C \tag{2.1}$$

, where D is diffusion coefficient(a scalar), and ∇C is the concentration gradient.

Since change in concentration is given by the negative divergence of the flux, we get the isotropic diffusion equation as above.

$$\frac{\partial C}{\partial t} = -\nabla \cdot F = \nabla \cdot D\nabla C \tag{2.2}$$

, where $\frac{\partial C}{\partial t}$: change of concentration in time.

In 3D isotropic media, scalar D is multiplied by identity matrix, then the equation becomes:

$$\frac{\partial C}{\partial t} = \nabla \cdot DI\nabla C \tag{2.3}$$

where we can name $DI = \begin{pmatrix} D & 0 & 0 \\ 0 & D & 0 \\ 0 & 0 & D \end{pmatrix}$ as the isotropic constant diffusion tensor in 3D.

Note that the corresponding 1D diffusion PDF is

$$\frac{\partial C}{\partial t} = D\frac{\partial^2 C}{\partial x^2} \tag{2.4}$$

The corresponding 1D diffusion along $e_1(\parallel e_1 \parallel = 1)$ with diffusion coefficient D, in 3D is given by

$$\frac{\partial C}{\partial t} = D{e_1}^T H_c e_1 \tag{2.5}$$

where H_c is the Hessian of the concentration. $({e_1}^T H_c e_1)$ is the second derivative of C along $e_1, C_{e_1 e_1}$.

For anisotropic media, total change in concentration can be formulates as the summation of three linear processed along orthogonal directions.

$$\frac{\partial C}{\partial t} = \lambda_1 C_{e_1 e_1} + \lambda_2 C_{e_2 e_2} + \lambda_3 C_{e_3 e_3} \tag{2.6}$$

$$\frac{\partial C}{\partial t} = \lambda_1 e_1^T H_c e_1 + \lambda_2 e_2^T H_c e_2 + \lambda_3 e_3^T H_c e_3 \tag{2.7}$$

where λ_i is the linear diffusion coefficient along $e_i(\parallel e_i \parallel = 1) e_i \perp e_j i \neq j$. Since $H_c = \nabla \cdot \nabla C$, we can write the above equation as

$$\frac{\partial C}{\partial t} = \nabla \cdot \lambda_1 e_1 {e_1}^T \nabla C + \nabla \cdot \lambda_2 e_2 {e_2}^T \nabla C + \nabla \cdot \lambda_3 e_3 {e_3}^T \nabla C \tag{2.8}$$

$$\frac{\partial C}{\partial t} = \nabla \cdot [\lambda_1 e_1 {e_1}^T + \lambda_2 e_2 {e_2}^T + \lambda_3 e_3 {e_3}^T] \nabla C \tag{2.9}$$

$$\frac{\partial C}{\partial t} = \nabla \cdot \left[\begin{pmatrix} e_1 & e_2 & e_3 \end{pmatrix} \begin{pmatrix} \lambda_1 & 0 & 0 \\ 0 & \lambda_2 & 0 \\ 0 & 0 & \lambda_3 \end{pmatrix} \begin{pmatrix} e_1 \\ e_2 \\ e_3 \end{pmatrix} \right] \nabla C \tag{2.10}$$

$$\frac{\partial C}{\partial t} = \nabla \cdot \mathbf{D} \nabla C \tag{2.11}$$

where **D** is the 3D diffusion tensor. Consider its eigen-decomposition.

$$\mathbf{D} = \begin{pmatrix} e_1 & e_2 & e_3 \end{pmatrix} \begin{pmatrix} \lambda_1 & 0 & 0 \\ 0 & \lambda_2 & 0 \\ 0 & 0 & \lambda_3 \end{pmatrix} \begin{pmatrix} e_1 \\ e_2 \\ e_3 \end{pmatrix} \tag{2.12}$$

We see that λ_i 's, the eigenvalues, correspond to linear diffusion coefficients along e_i's, the eigenvectors. Thus we get

$$C_t = \nabla \cdot \mathbf{D} \nabla C \tag{2.13}$$

$$M = \int_{-\infty}^{+\infty} C \partial\Omega \tag{2.14}$$

where M is the amount of substance and $\partial\Omega$ is unit volume. r is the position vector that has components(x,y,z). We can define the probability distribution function(PDF) of the spatial distribution of diffusion particles at time t, which were initially at the origin at t=0 , as

$$P(t, r) = \frac{C(t, r)\partial\Omega}{M} \tag{2.15}$$

$$P(t, r) = \frac{1}{\sqrt{(4\pi D t)^3}} exp[-\frac{\|r\|^2}{(4\mathbf{D}t)}] \tag{2.16}$$

This equation is zero mean Gaussian PDF with variance $\sigma^2 = 2Dt$ along all directions. This equation can also be written as

$$P(r) = \frac{1}{\sqrt{(2\pi)^3 \Delta_c}} exp[-\frac{1}{2} r^T \mathbf{C}^{-1} r] \tag{2.17}$$

where C is the covariance matrix and Δ_c is the determinant of C and

$$C = \begin{pmatrix} 2Dt & 0 & 0 \\ 0 & 2Dt & 0 \\ 0 & 0 & 2Dt \end{pmatrix} \tag{2.18}$$

We can express the spatial distribution PDF of diffusing particles which were originally at the origin as

$$P(r) = \frac{1}{\sqrt{(2\pi)^3 \sigma_1 \sigma_2 \sigma_3}} exp[-\left(\frac{r_1^2}{2{\sigma_1}^2} + \frac{r_2^2}{2{\sigma_2}^2} + \frac{r_3^2}{2{\sigma_3}^2}\right)] \tag{2.19}$$

$$P(r,t) = \frac{1}{\sqrt{(2\pi)^3 2^3 (\Delta \mathbf{D} t)^3}} exp[-\left(\frac{1}{2}\frac{r^T \mathbf{D}^{-1} r}{2t}\right)] \tag{2.20}$$

where $\sigma_i^2 = 2\lambda_i t$ is the variance along e_i, $\frac{D^{-1}}{2t}$ is the covariance matrix and $\Delta\mathbf{D}$ is the determinant of the covariance matrix. It was shown that the solution of this PDF for an initially point source(concentration) i.e. for $C(r, t = 0) = \delta(r)$, is

$$P\left(r,t\right) = \frac{1}{\sqrt{(4\pi t)^3 |\mathbf{D}|}} * e^{\frac{\left(-\mathbf{r}^T \mathbf{D}^{-1} \mathbf{r}\right)}{4t}} \tag{2.21}$$

The above equation is a 3D multivariate gaussian. A direct interpretation of is is to consider it as the PDF of the distribution of diffusion particles. This is a zero mean PDF with covariance matrix given by 2tD.

2.1.3. Diffusion Tensor

In the previous section, we have explained how diffusion is defined by using a symmetric semi-definite second order matrix. This matrix can be analyzed using its eigen decomposition. It can also be analyzed as the covariance matrix of 3D multivariate gaussian function.

Table 2.1. Important Parameters that characterizes the DTI

Mean diffusivity $< D >= \frac{Tr(\mathbf{D})}{3} = \frac{(\mathbf{D}_{xx}+\mathbf{D}_{yy}+\mathbf{D}_{zz})}{3} = \frac{(\lambda_1+\lambda_2+\lambda_3)}{3}$
Fractional Anisotropy FA $FA = \sqrt{\frac{3(L_1^2+L_2^2+L_3^2)}{2(\lambda_1^2+\lambda_2^2+\lambda_3^2)}}$, where $Li = \lambda_i - < \mathbf{D} >$
Anisotropy Shape: $\lambda_1 > \lambda_2 > \lambda_3$ corresponding eigenvectors e_1, e_2 and e_3
Linear coefficient:
$cl = \frac{(\lambda_1-\lambda_2)}{(\lambda_1+\lambda_2+\lambda_3)}$
Planar coefficient:
$cp = \frac{2*(\lambda_2-\lambda_3)}{(\lambda_1+\lambda_2+\lambda_3)}$
Spherical coefficient:
$cs = \frac{3*\lambda_3}{(\lambda_1+\lambda_2+\lambda_3)}$
and $c_l + c_p + c_s = 1$

By using eigenvalues and eigenvector, some parameters can be calculated, and these are parameters can be used represent the diffusion tensor. In table Table 2.1, formulation of these parameters are shown.

Mean diffusivity is a measure of the diffusion in a voxel, free of anisotropy and it provides a value that is invariant with tissue direction or orientation.

Anisotropy is the property of presenting different characteristics depending on the orientation. Fractional Anisotropy FA, is a type of measure of anisotropy. It is definition of ellipsoid eccentricity.

Anisotropy shape is defined using these coefficients(c_l, c_p, c_s) which are based on the eigenvalues of a tensors. These three coefficient tell us the type of diffusion at that point. If c_l is bigger, this means λ_1 is significantly different than the other two eigen values. In this case, diffusion is linear and strong at the principle diffusion direction(e_1). If c_p is bigger, this means that λ_1 and λ_2 are closer to each other and they are much larger than λ_3. In this case isotropy is planer. If c_s is bigger, three eigen values are closer to each other. That is the tensor is isotropic.

2.1.4. Tensor Visualization

Tensor Visualization is a very active research field nowadays, due to the large number of application areas. But, because of the multi-variate nature of individual tensor samples, tensor field visualization is a challenging task. Designing a user interface is also difficult, given that humans are not accustomed to analyze data with so many dimensions. Moreover, visual representation is highly application specific. In geomechanics, tensors are used to show the response of material to applied forces. In flow analysis, gradient tensor of a flow field represents the structure of the flow. In medical imaging, diffusion tensor is used to describe the directional dependence of molecule motion. In DTI application, tensor field represents the motion of water molecules in a biological tissue by the action of Brownian motion.

In this section, common tensor field visualization methods are examined. Techniques that are used in our tensor visualization tool are discussed in detail. These are limited with 3D second order symmetric tensor fields.

2.2. Techniques

2.2.1. Object Based Techniques

These techniques are characterized by representing tensors or their properties by geometrical objects.

1. Glyphs

   A common method to represent a tensor at each point of a tensor field is to use glyphs. Glyphs are parameterized icons that represents the data by its shape, color, texture, location, etc. Over the years, a variety glyphs have been designed for tensor visualization. They can be classified by tensor characteristics that are used in definition of the glyph shape. Main characteristics of tensor are eigenvalues and eigenvectors; anisotropy coefficients, and other measures like curvature, shear, rotation etc.

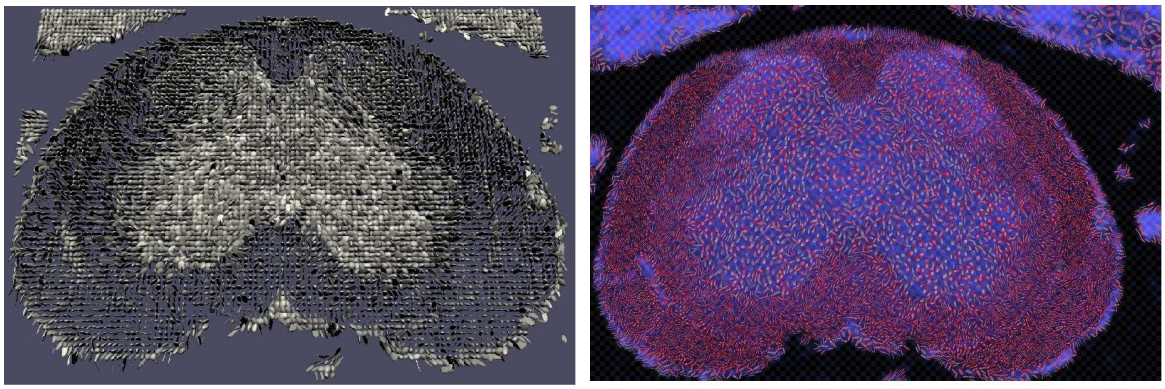

Figure 2.1. a)Normalized ellipsoids to visualize diffusion tensor field in a single slice and b) With coloring to illustrate direction and magnitude of diffusion[4]

a Eigenvalues and eigenvector: If eigenvalue and eigenvector are used for tensor field visualization, ellipsoids are naturel choice for a glyph to summarze the information contained in a diffusion tensor. The three radii are proportional to the eigen values. And its orientation is determined by the three orthogonal eigenvectors of tensor. Its major axis is aligned with the major eigenvector, and medium and minor axes are aligned with the medium and minor eigenvectors, respectively. This representation can be used only for symmetric matrices, because eigenvalues should be real and nonnegative and eigenvectors should be real and orthogonal.

   If λ_1 is bigger that other two eigenvalues, ellipsoid will be linear in the major eigenvector direction. If λ_1 and λ_2 are similar, but bigger than λ_3, shape is like a disk(planar). If three lambda values are similar, shape will be spherical(isotropic).

   Laidlaw normalized the size of the ellipsoids to fit a large number of them in a single image.[4] While this eliminates the ability to show mean diffusivity, it creates more uniform glyphs which generates a better visualization of anatomy and pathology. Color is used to provide information lost in the normalization(Figure 2.1).

b Anisotropy Coefficients: There are three main methods, when the anisotropy

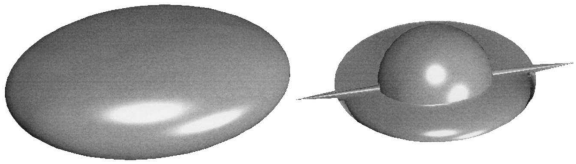

Figure 2.2. Composite glyphs[5]

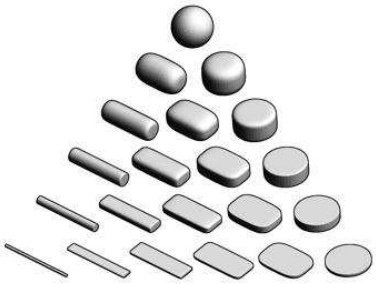

Figure 2.3. Superquadrics[6]

coefficients are used for glyph creation:

i. Composite

Westin used a composite of linear, planar and spherical components of diffusion tensor.[5] Arrow, disk and sphere glyphs represent the coefficients c_l, c_p, and c_s. Each shape is proportional to these coefficients and assigned a different color. Arrows are colored blue, disks are colored yellow and spheres are colored red as shown in Figure 2.2.

ii. Superquadrics

Kindlmann adapted traditional surface-modeling technique called superquadrics as a glyph.[6] As shown in Figure 2.3, in linear and planar cases, anisotropy is emphasized by using cylinder. Sphere is used in isotropic case. Superquadric tensor glyphs have the symmetry properties of ellipsoids, while also imitating cuboids and cylinders to better convey shape and orientation, where appropriate.

Glyph Packing is a new method for using glyphs in tensor field visualization.[7]. In Figure 2.4, we can see the difference between normal glyph representation and glyph packing.

c Other Parameters

Wijk used other information about the curvature, share, velocity, rotation, acceleration and convergence/divergence.[8] This is called flow probe and

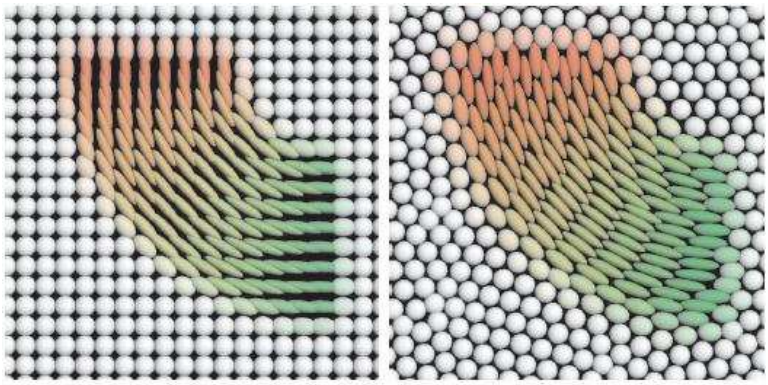

Figure 2.4. Glyph Packing[7]

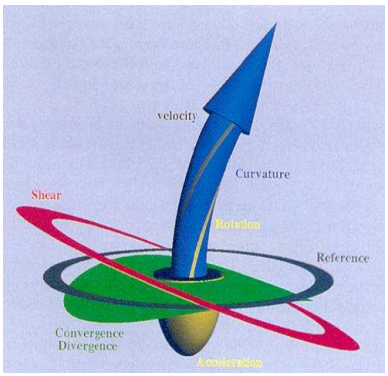

Figure 2.5. Flow Probes[8]

illustrated in Figure 2.5.

2. Tractography

   Tractography yields continuous pathways that are not possible to show using discrete glyphs. Here, basic tensor visualization methods that are in the class of tractograpy are explained.

   a Hyperstreamlines

   Delmarcelle and Hesselink proposed streamlines as a representation of tensor fields[10]. Each vector in the field is tangent to a streamline in a streamline representation. Hyperstreamline is similar streamline but its shape is a tube with an elliptical cross-section. Tube direction is determined by principle eigenvector and size of elliptical cross-section is determined by medium and minor eigenvalues. Hyperstreamlines provide continuity. Coloring can be added to show the linear anisotropy or other properties.Figure 2.6

   b Streamtubes and Streamsurfaces

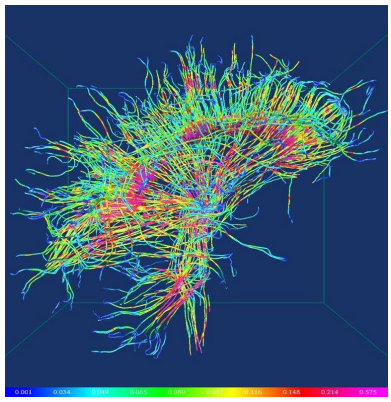

Figure 2.6. Hyperstreamlines[9]

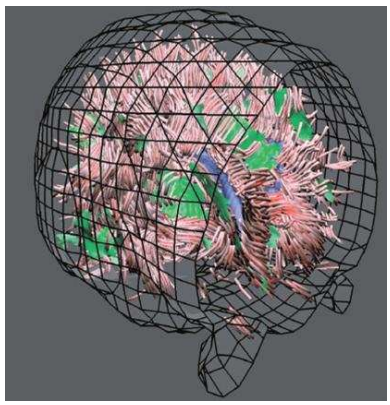

Figure 2.7. Streamtubes and streamsurfaces[11]

Zhang proposed streamtubes to overcome defects of streamlines.[11, 12] First of all, its cross-section can get large, this can cause visualization of less hyperstreamlines. Secondly, hyperstreamlines can not visualize regions where anisotropy is mainly planar. In this method, regions of linear and planar anisotropy are split. Linear isotropy is represented by streamtubes, and planar anisotropy is represented by streamsurfaces. But, the threshold should be determined properly. Hyperstreamtubes is very similar to streamlines. The difference is that the major axis of cross-section is forced to be constant but the proportion is the same. And color is used to solve the ambiguity. Streamsurfaces are represented by plane that is determined by major and medium eigenvector.Figure 2.7.

c Techniques of Streamline Extraction

The standard streamline technique advects massless particles through the vector field and traces their location as a function of time. Analogously,

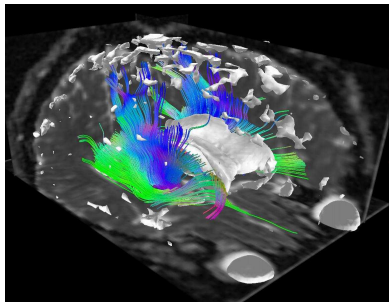

Figure 2.8. Streamline Extraction[13]

a hyper-streamlines approach has been proposed to trace changes through tensor fields, following the dominant eigenvector direction. These methods work best on very clean datasets, which are usually produced as a result of simulations; these methods typically do not handle raw experimental data very well, due to noise and resolution issues.

Although these methods provide nice visual cues, they do not attempt to recover the underlying anatomical structures, which are the white matter fiber tracts (bundles of axons) found within the brain. Several previous endeavors have been made for recovering the underlying structure by extracting fibers through the application of modified streamline algorithms. Tensorlines methos is one of these efforts.[21],Figure 2.9.

Zhukov et al. have develop a new technique for tracing anatomical fibers from 3D tensor fields. The technique extracts salient tensor features using a local regularization technique that allows the algorithm to cross noisy regions and bridge gaps in the data. They applied the method to human brain DT-MRI data and recovered identifiable anatomical structures that correspond to the white matter brain-fiber pathways.[13]

2.2.2. Texture Based

a LIC and HyperLIC

On vector field visualization area, there are some conflicting but desirable criteria : accuracy, locality of calculation, simplicity, controllability and generality. Line Integral Convolution(LIC) is a technique that includes many of these criteria.

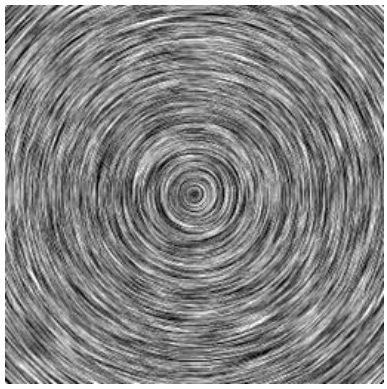

Figure 2.9. LIC Output Image[16]

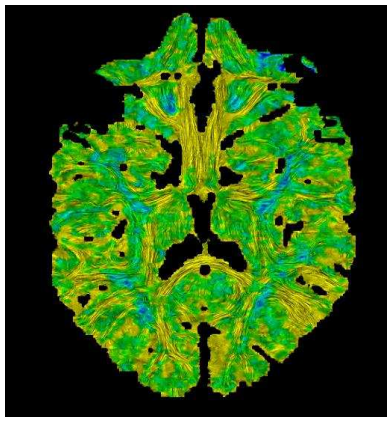

Figure 2.10. DT-MRI using HyperLIC[15]

LIC can be analyzed in 2D, and it can be extended to 3D by using the same calculations in third dimension. LIC has two main steps. One is local stream line generation and the other is the convolution.

As a first step, the local behavior of the vector field can be approximated by computing a local streamline for each pixel. Each streamline starts at the center of a pixel and moves out in positive and negative directions.

After local streamlines are calculated, noise in pixels that streamlines goes are convolved using a weight function. And output value is used as a new pixel value. The resultant 2D image is given in Figure 2.9.

HyperLIC is a technique that fits better tensor visualization. In this method, tensor is decomposed to eigenvalues and eigenvectors. Main difference of this method from LIC is to use three eigenvector fields. LIC method is used to visualize DTI data[17, 18] and several tensor field[19].

b Reaction-Diffusion Textures

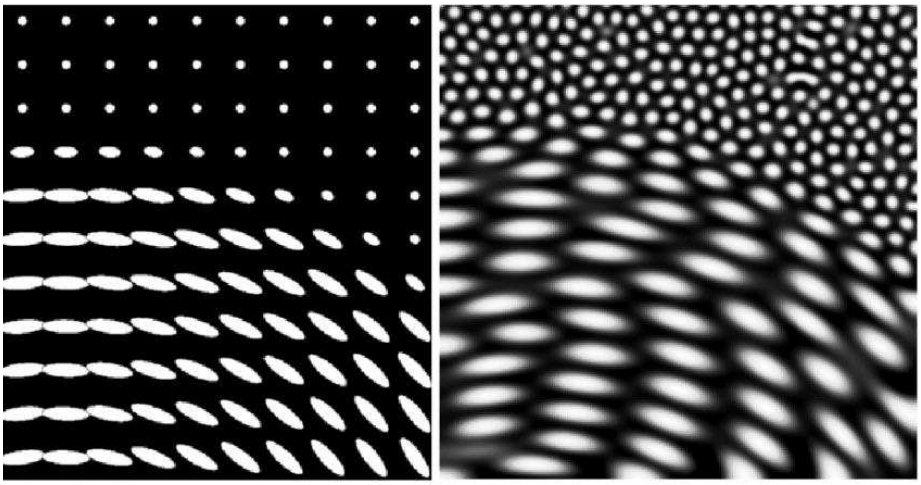

Figure 2.11. Reaction-diffusion texture[14]

In HyperLIC, a white noise image are modified according to characteristics of given vector field. In reaction-diffusion textures, pattern of ellipsoids is used instead of noise.[15]

A reaction-diffusion system models the reactions between two chemicals, which diffuse at different rates and interact according to certain rules of activation and inhibition , and may be employed for texture generation in computer graphics. Kindlmann modified the model for the case of a single chemical for use in tensor field visualization.[14]

2.2.3. Volume Rendering

a Barycentric Map

Important point in visualization is to show relevant information to the user and to hide unrelevant part. To achieve this for diffusion tensor visualization, opacity value is used in volume rendering. Data is mapped by a transfer function, which must be carefully designed to delineate and emphasize the features of interest. Data having greater interest are rendered more opaque and others are rendered transparent.

Kindlmann propose a function based on a barycentric opacity map in Figure

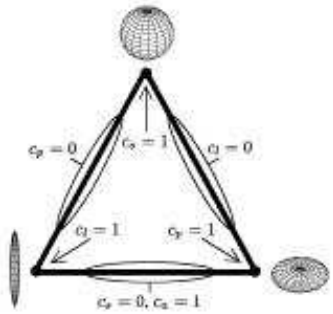

Figure 2.12. Barycentric Map with Tensor anisotropy coefficients[22]

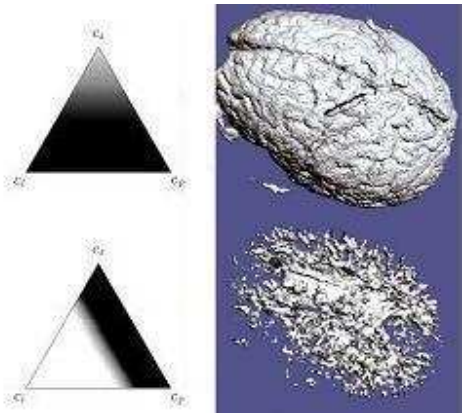

Figure 2.13. Barycentric maps and the resulting volumes[22]

2.12. In this map, opacity depends on the anisotropy coefficients c_l, c_p, and c_s. Barycentric map is constructed by assigning transparency values for each point in the equilateral triangle. Opacity value for each tensor will determined by c_l, c_p and c_s values of this tensor. [22].

In Figure 2.13, we can observe an example of volume rendering of DTI data for specific barycentric map. Color can be added this representation. In figure 2.14, a DTMRI is visualized by using barycentric maps that determine color and opacity given to a tensor. Using color helps to differentiate the regions of linear and planar anisotropy.

deneme

b Hue-Balls and Lit-Tensors

Kindlmann and Weinstein proposed two methods that are called Hue-Balls and Lit-Tensor for diffusion tensor visualization.[23]

In HueBalls methods, a color map on a sphere determines the vector orientations.

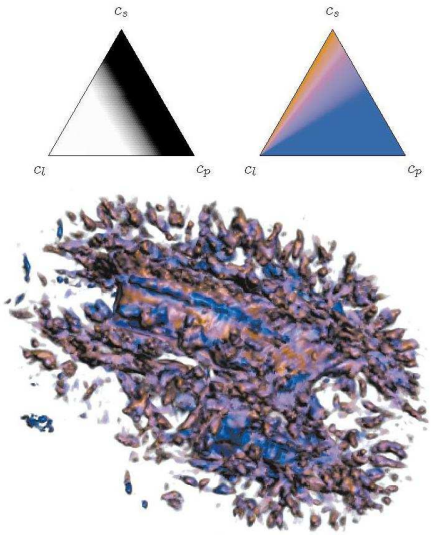

Figure 2.14. Volume rendering using barycentric maps for tensor opacity and color[22]

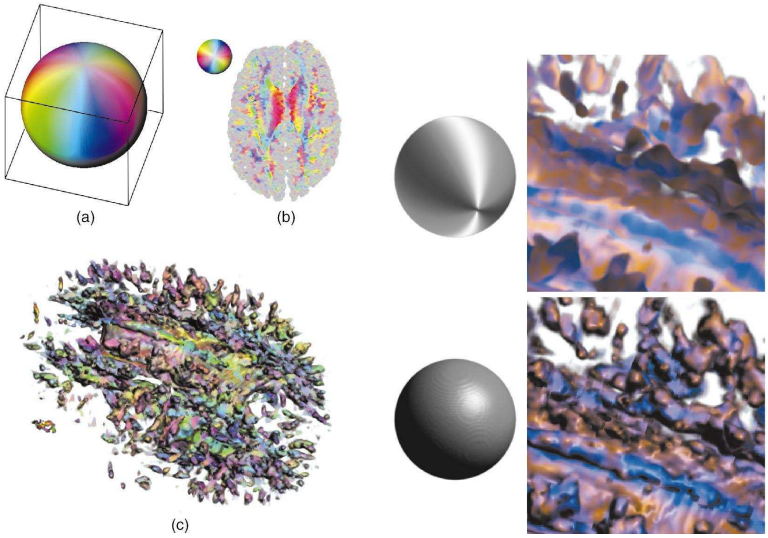

Figure 2.15. Hueball method

This representation provides continuous mapping from direction to color. Lit Tensors are a shading technique for tensors that also allows a better appreciation of the type and orientation of their anisotropy. In figure 2.15, left image is an example rendering of DT-MRI volume with hue-balls. In right image, we can observe the difference by using lit-tensor.

3. DIFFUSION TENSOR FIELD VISUALIZATION TOOL (DTVT)

Tensor visualization is highly application dependent. The information that tensor data represents and requirements of the application plays an important role on the technique selected for visualization. In this thesis, only the visualization of diffusion tensor fields are considered. Therefore, the investigation of visualization requirements of diffusion tensor field are provided.

The methods in the thesis are tested by using a phantom diffusion tensor field. Therefore the phantom data are explained before the representation of methods. Firstly, basic visualization methods that are mentioned in literature survey are applied to diffusion tensor field. After that, we propose two new methods, that tries to fulfill the requirements that are provided in here. First method based on the physical and mathematical meanings of diffusion tensor that are introduced in chapter 2. Also, parameters that characterizes diffusion are used in visualization. At the end of the chapter, user interface of the tool is described.

3.1. Requirements of Diffusion Tensor Field Visualization

Connectivity and the flow path are the key concepts, that play important role in the interpretation.[24]. Connectivity is a representation of the diffusion between neighboring voxels. Connectivity maps are the networks that includes the connectivity information. We form four types connectivity maps to analyze and visualize the diffusion. Flow path can be defined as a structure that shows the diffusion on the whole volume. The connection that shows diffusion between any two points in the volume is a flow path. Any visualization technique should incorporate one or both of these concepts.

- Connectivity Visualization

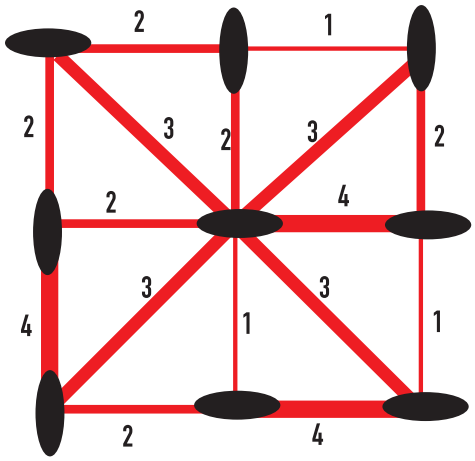

Figure 3.1. Connectivity with neighbor

For connectivity visualization, we have used a connectivity map where connections between neighboring tensor pairs are weighted by scalar constants. Connectivity map depend on the local connectivity definition used possible parameters. A successful visualization requires the diffusion region to be easily differentiated . Connectivity map should provide high connection values when there is diffusion and low values when there is no diffusion. Figure 3.1 shows different combinations of tensor pairs with connectivity values represented via the thickness. Pairs that have high connectivity are shown to be connected with thick lines and pairs that have low connectivity are shown to be connected with bold lines.

- Flow Patterns Visualization

  Flow pattern approaches attempt to visualize the overall flow structure within the volumetric diffusion field. A texture map or an image is modulated as a function of the principle direction of each diffusion tensor. Paths and patterns of diffusion are extracted by using the vector field that is principle diffusion direction of diffusion tensor field. They are also are reconstructed by using numerical integration in diffusion direction, the most popular one is 4th order Runge-Kutta, Figure 3.2. Flow pattern based approach also allow the visualization of singularities in vector field that inform us about the characteristics of flow. Saddle Points, Focus and Nodes are examples of these singularities. Figure 3.3

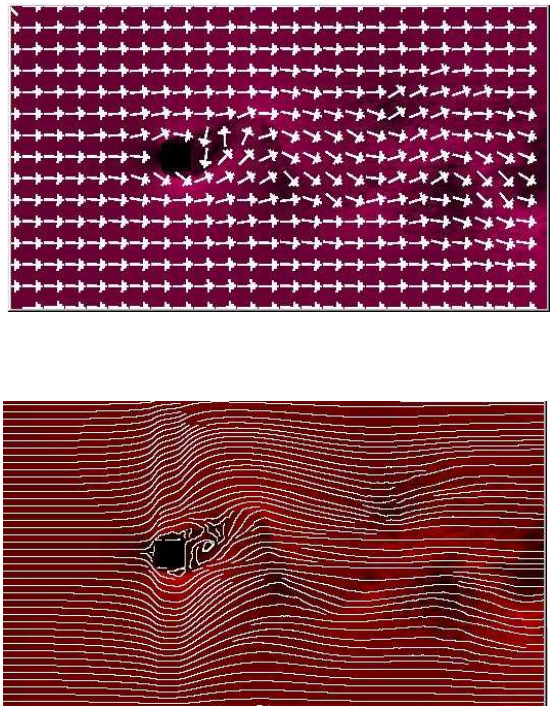

Figure 3.2. Flow Vector Field and Streamline interpolation by using Runge-Kutta [33]

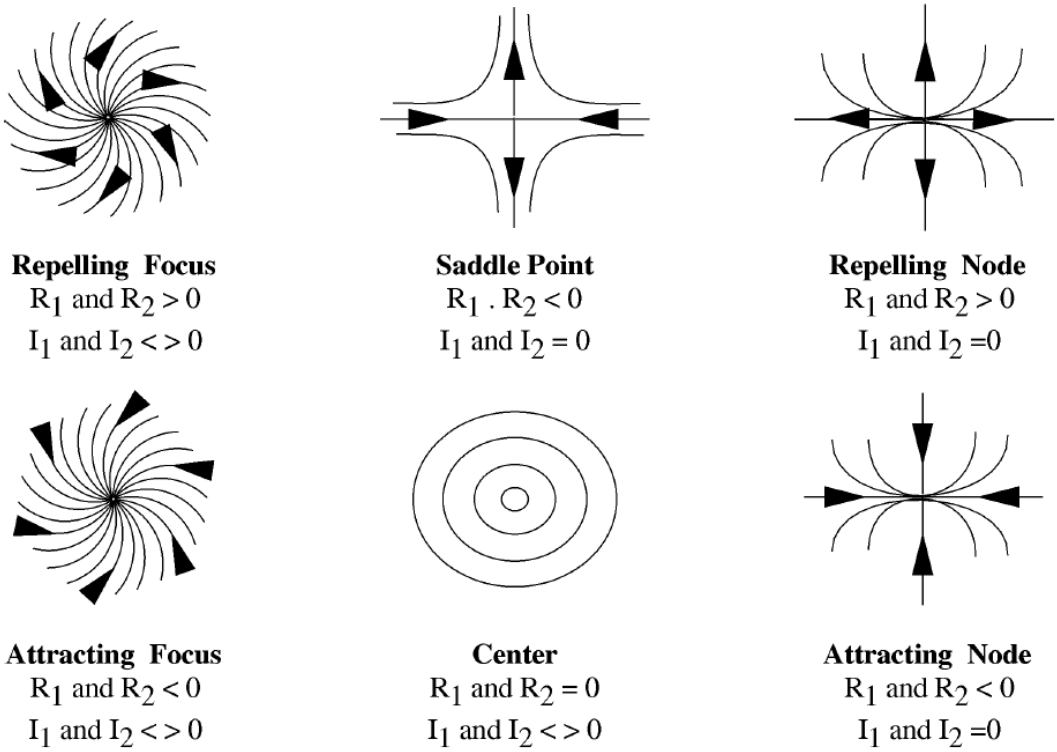

Figure 3.3. Sink, Sources and Degenerate Points[20]

Figure 3.4. Geometry of the diffusion tensor phantoms: Rings[25]

3.2. Phantom Diffusion Data

Two phantom diffusion tensor fields with different SNR-values are used for the evaluation of methods. There are two sets of data with SNR values 32 and 8. Geometry of diffusion tensor field is shown in Figure 3.4, it is composed of five concentric fiber rings with different curvature values. The noiseless FA-value set for high diffusion regions is 0.82 and for low diffusion regions is 0.13. The maximum resolution is 0.1 unit and diameters of rings is 1 unit.

The DW-MR signals were acquired for diffusion-encoding along 6 directions and one base signal. Independent Gaussian noise ($\sigma = 1/SNR$) is added to real and imaginary parts of each complex MR signal. Subsampling was done by truncation in k-space followed by Hamming window smoothing. Final MR signals were used to compute diffusion tensor fields. The data was re-sampled at 0.1 units at the end of B-Spline interpolation and approximation schemes. Any such re-sampling for the linear scheme haven't applied. [25]

3.3. Basic Visualization

Initially, a set of well-known tensor visualization methods that are explained in Chapter 2 are employed to investigate the properties of diffusion tensor field data. In these methods, mathematical properties of each tensor are visualized individually. Relationships between neighboring tensor pairs are ignored.

Firstly, eigen-decomposition interpretation of diffusion tensor is used. For each

tensor, an ellipsoid(glyph) is generated using the eigenvalues and eigenvectors of the tensor. Eigenvalues determine the three radii of the ellipsoid in three dimensions and eigenvectors determine the orientation of the ellipsoid.

Here, if λ_1 is bigger that other two eigenvalues, ellipsoid will be linear in the major eigenvector direction. This indicates that diffusion is linear. If λ_1 and λ_2 are similar, but they are bigger than λ_3, the shape is like a disk. This type of tensors are shown where more than one diffusion areas are crossing each other or where a diffusion area bifurcates. If three lambda values are similar, the shape will be spherical which means that there is no diffusion because of isotropy.

To color the tensor field for visualization, two methods are used. Firstly, three components of the principle eigenvector are used for coloring. Red, green and blue components of color are determined according to x, y and z components of first eigenvector, respectively [26].
R = abs(PDD_x),
G = abs(PDD_y),
B = abs(PDD_z)

As a result of this coloring method, tensors that are aligned in x direction are colored red, those aligned in y direction are colored green, those aligned in z direction are colored blue. Sign of eigenvectors are not defined. Therefore, two opposite vectors can be colored identically.

Secondly, diffusion anisotropy metrics are used. As we know, c_l, c_p, and c_s are measures for linear, planar, and spherical diffusion respectively [27].
$R = c_l = (\lambda_1 - \lambda_2)/\lambda_1$,
$G = c_p = (\lambda_2 - \lambda_3)/\lambda_1$,
$B = c_s = \lambda_3/\lambda_1$,
where $\lambda_1, \lambda_2, \lambda_3$ are eigenvalues and $\lambda_1 > \lambda_2 > \lambda_3$

As a result of this coloring method, linear tensors will be in color red, collinear

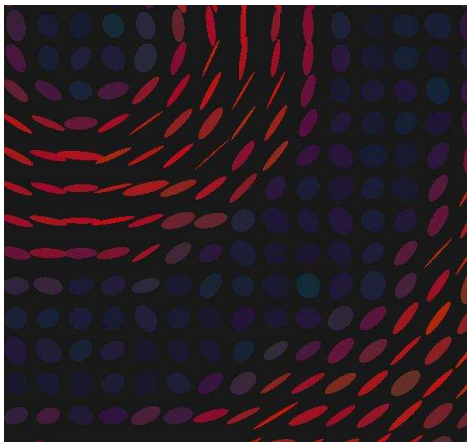

Figure 3.5. Detailed picture of diffusion tensor field data with SNR32. Each tensor is visualized with an ellipsoid and colored using the components of the first eigenvector.

tensors will be in color green, and spherical tensors will be in color blue.

In addition to the use of color in visualization, variations in opacity of glyphs can be used to visualize other properties such as the fractional anisotropy of the tensor. This is shown in Figure 3.5. Tensors that have high FA values are more opaque and those that have low FA values are more transparent. In Figure 3.6, the results of two basic coloring methods are shown for datasets with different SNR-values.

3.4. Connectivity Analysis Approach: Tensor Painting

In the technique, we have named tensor painting, the aim is to gradually paint glyphs, that define diffusion tensors as growing regions, using the connectivity between tensor pairs. By spreading paint in the tensor field, the diffusion is visualized.

Tensor painting is based on two components. These are local connectivity definition and threshold. Local connectivity definition tries to measure the strength of diffusion between tensor pairs. Then, comparing the threshold by this value, painting decisions are made and local connectivity paths are created.

Tensor Painting is summarized in block diagram in Figure 3.7. Before painting starts, all local connectivity values are calculated. Then, a seed region are selected

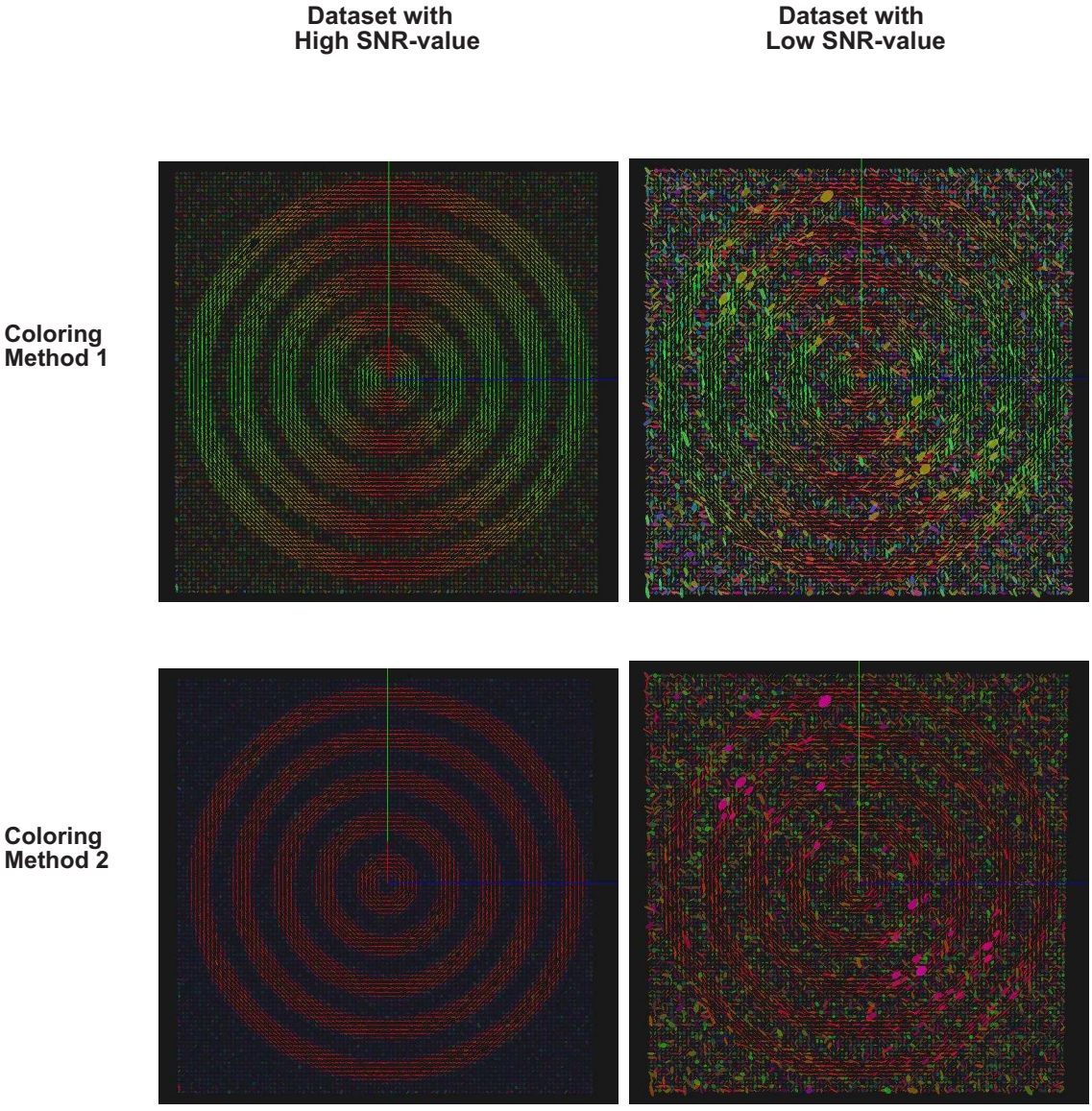

Figure 3.6. Basic coloring in datasets with low and high SNR-values.

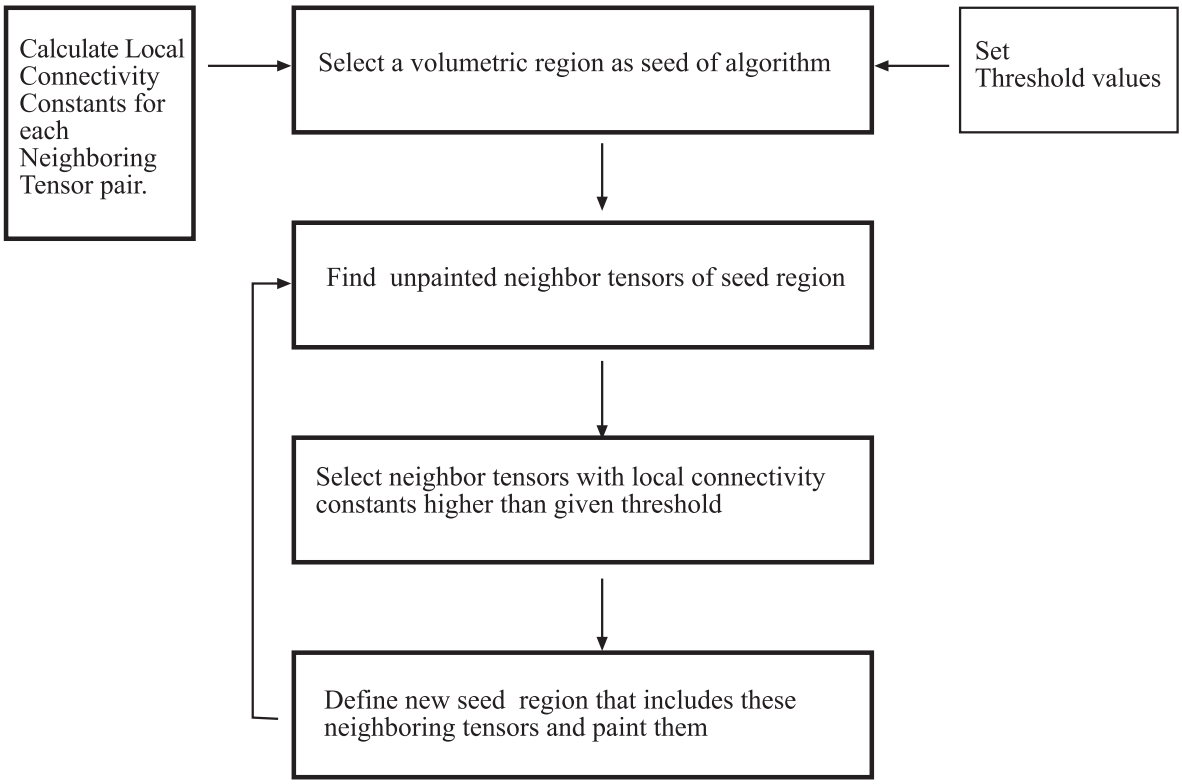

Figure 3.7. Tensor Painting

and tensors in this region are painted. All neighbors of the tensors in this region are determined. And neighbors whose local connectivity values are higher than the determined threshold are found. These tensors are defined as new seed regions and painted. This process is repeated for this new seed tensors.

In the Tool, by coloring painted tensors a stepwise fashion, an animation affect is created (Figure 3.8). The animation visualizes the diffusion spreading from selected region in tensor field. Paint tool allows the modification of local connectivity parameters by editing the principle eigenvalue of all tensors. Changes in local connectivity and thresholds affects the visualization.

In figure 3.9, the idea behind the local connectivity definition is explained by using four basic tensor pair configurations. In first case, the diffusion is high because the tensor directions are the same. In second and third case, there is medium diffusion because directions of two tensors are orthogonal to each other. And in the last case, two tensors are parallel to each other, so diffusion is very low. Using connectivity

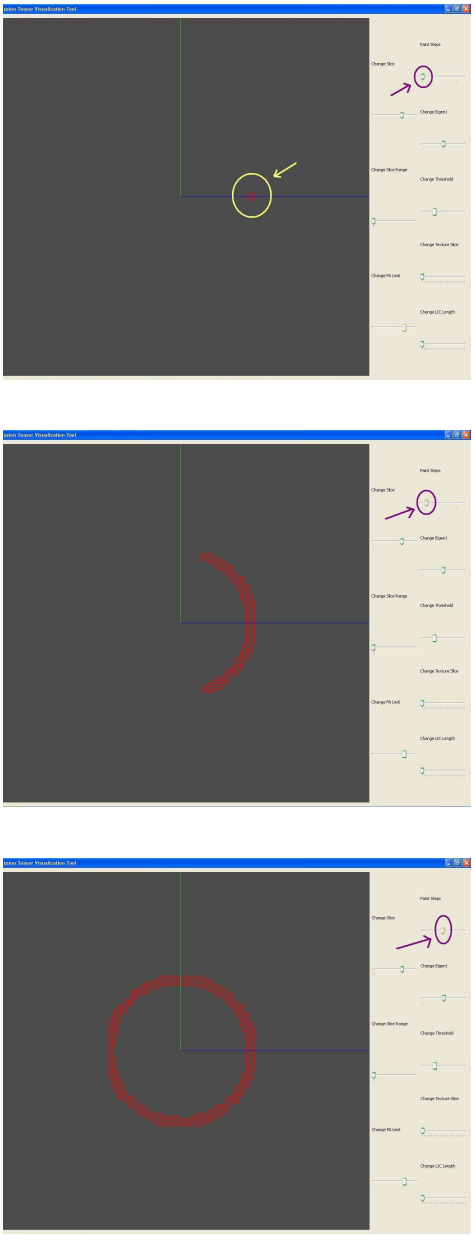

Figure 3.8. Coloring Painted Tensors A Stepwise Fashion

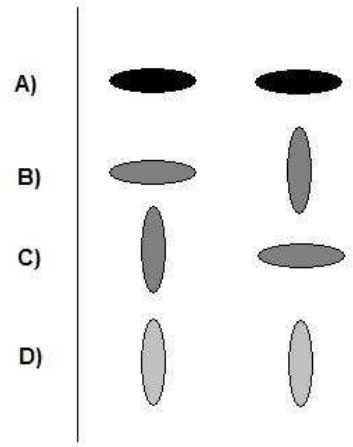

Figure 3.9. Four basic tensor pair configurations

analysis, we assign connectivity constants to tensor pairs by proportionally using these basic evaluations.

To achieve this, four connectivity definitions are proposed by using different properties of diffusion tensors. They are all based on interpretation that each diffusion tensor($\mathbf{D}$) determines the covariance matrix of the 3D Gaussian PDF of the distribution of diffusing particles which are initially located at the origin at t=0. This PDF is given as,

$$f(\mathbf{r},t) = \frac{1}{\sqrt{(4\pi t)^3|\mathbf{D}|}} * e^{\frac{\left(-\mathbf{r}^T\mathbf{D}^{-1}\mathbf{r}\right)}{4t}} \tag{3.1}$$

where $\mathbf{r} = [x, y, z]$ is the 3D position vector, t is the time instant where the observation is made and $\mathbf{D}$ is the diffusion tensor. ${\sigma_i}^2 (i = x, y, z)$ is related to the diffusion tensor as ${\sigma_i}^2 = 2\lambda_i t$ where λ_i is the eigenvalue of $\mathbf{D}$ corresponding to the dimension i. Setting t=1 (arbitrary choice), we can write the 3D distribution of an diffusing particle that was at the origin at t=0, as

$$\tilde{f}(\mathbf{v}_{ji}) = \frac{1}{\sqrt{(4\pi)^3|\mathbf{D_j}|}} * e^{\frac{\|\mathbf{v}_{ji}\|^2\left(-\hat{\mathbf{v}}_{ji}^T{\mathbf{D_j}}^{-1}\hat{\mathbf{v}}_{ji}\right)}{4}} \tag{3.2}$$

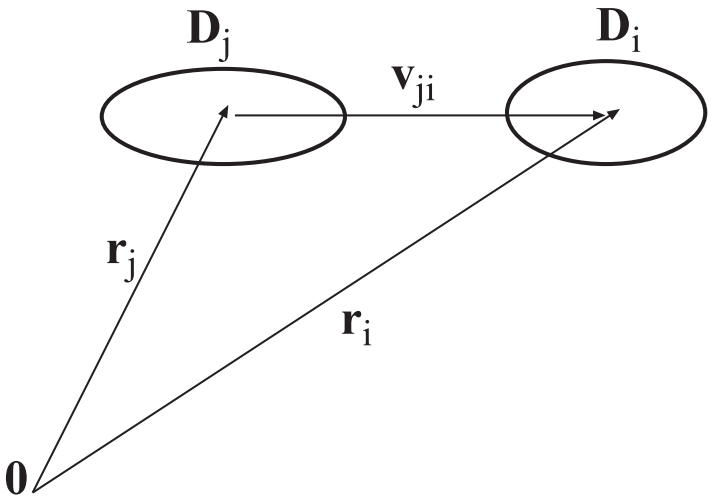

Figure 3.10. Notation for 3D gaussian PDF

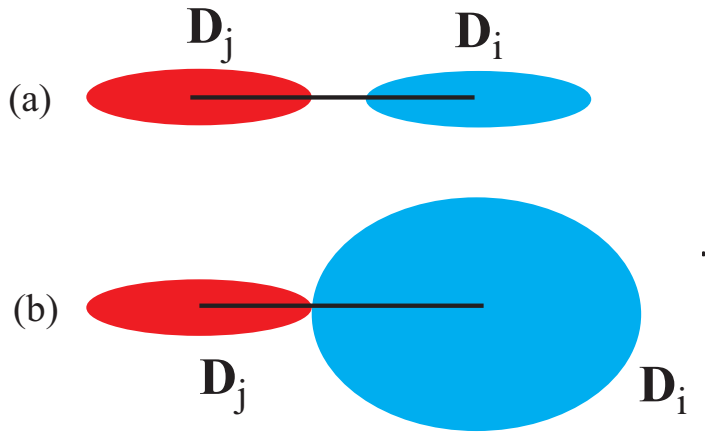

Figure 3.11. Effect of lack of normalization in isotropy

where $\mathbf{v}_{ji} = \parallel v_{ji} \parallel * \hat{v}_{ji}, |\mathbf{D}_j| = \lambda_{xj}\lambda_{yj}\lambda_{zj}$

As seen in Figure 3.10, $\mathbf{r}_j$ is used as the position vector for tensor j($\mathbf{D}_j$) in 3D tensor field. $\mathbf{v}_{ji}$ is position vector neighbor of $\mathbf{D}_i$ with respect to $\mathbf{D}_j$

In all proposed connectivity definitions, our aim is to obtain a connectivity measure between neighboring tensors, proportional to the mutual diffusion between them.

Defining the mutual diffusion based on the 3D PDFs only may be misleading. Such a definition is affected by the total diffusion whereas we are interested in the anisotropic diffusion which is known to correlate with fiber structure(connectivity). Figure 3.11 depict two cases. The local connectivity based on 3D PDFs only turns out

to be higher for the cases in Figure 3.11 b than the case in Figure 3.11 a. However, Figure 3.11 a corresponds to a higher structural connectivity. The effect of the high isotropic diffusion in Fig 3.11 b is needed to be normalized. Normalization can be achieved by removing the effect of isotropy and using only the anisotropic(deviatoric) part of a diffusion tensor.

To achieve this, Let us decompose a 3×3 2^{nd} order tensor $\mathbf{T}$ as,

$$T = \bar{T} + \tilde{T} \tag{3.3}$$

where $\bar{T}$ represents the isotropic part of the tensor and $\tilde{T}$ represents the anisotropic (deviatoric) part of the tensor. Each part is defined as follows,

$$\bar{\mathbf{T}} = \frac{1}{3} trace\,(\mathbf{T})\, I \tag{3.4}$$

$$\tilde{\mathbf{T}} = \mathbf{T} - \frac{1}{3} trace\,(\mathbf{T})\, I \tag{3.5}$$

Considering the eigen-decomposition interpretation of the diffusion tensor D as,

$$D = \begin{pmatrix} \mathbf{e}_1 & \mathbf{e}_2 & \mathbf{e}_3 \end{pmatrix} * \begin{pmatrix} \lambda_1 & 0 & 0 \\ 0 & \lambda_2 & 0 \\ 0 & 0 & \lambda_3 \end{pmatrix} * \begin{pmatrix} \mathbf{e}_1{}^T \\ \mathbf{e}_2{}^T \\ \mathbf{e}_3{}^T \end{pmatrix}$$

the deviatoric part is given as,

$$D' = \begin{pmatrix} \mathbf{e}_1 & \mathbf{e}_2 & \mathbf{e}_3 \end{pmatrix} * \begin{pmatrix} \lambda_1 - \alpha & 0 & 0 \\ 0 & \lambda_2 - \alpha & 0 \\ 0 & 0 & \lambda_3 - \alpha \end{pmatrix} * \begin{pmatrix} \mathbf{e}_1{}^T \\ \mathbf{e}_2{}^T \\ \mathbf{e}_3{}^T \end{pmatrix}$$

where $\alpha = (\lambda_1 + \lambda_2 + \lambda_3)/3$

When the average of eigenvalues is subtracted from each individual eigenvalue of the tensor, minimum eigenvalue will be negative. Median eigenvalue can also be negative. This is not meaning for a diffusion tensor. Moreover, it may produce wrong explanations for diffusion. This can be corrected by adding minimum eigenvector to all three or by setting all negative values to zero. By doing so, we can get nonnegative eigenvalues, but zero eigenvalues still can cause some division problems. To solve this problem, a constant is added to all eigenvalues. After eigenvalues are updated, diffusion tensor is recalculated. Eigenvectors of tensor are not affected. Figure 3.12 shows the two dataset with different SNR-values and same datasets after the two types of deviatoric is applied.

3.4.1. Geometrical Connectivity Between Diffusion Tensors

The first connectivity definition we have used, is based on the geometrical distance measure between neighboring ellipsoids that represent the respective tensors. The iso-probability surfaces set at $K_{ji}\tilde{f}(\mathbf{v}_{ji} = 0) with K_{ji} = 0.9$ of the 3D Gaussian PDFs corresponding to those tensors.

Follows Equation 3.2, K_{ji} is given as,

$$K_{ji} = e^{\frac{\left(-\hat{\mathbf{v}}_{ji}^T {\mathbf{D}_j}^{-1} \hat{\mathbf{v}}_{ji}\right) {d_{ji}}^2}{4}} \tag{3.6}$$

where d_{ji} is the distance from center of $\mathbf{D}_j$ towards $\mathbf{D}_i$ where $\tilde{f}(.) = K_{ji}\frac{1}{\sqrt{(4\pi)^3|\mathbf{D}_j|}}$ d_{ji} can be solved as,

$$d_{ji} = \sqrt{-\frac{4\ln(K_{ji})}{\left(\hat{\mathbf{v}}_{ji}^T {\mathbf{D}_j}^{-1} \hat{\mathbf{v}}_{ji}\right)}} = \sqrt{-\frac{4\ln(0.9)}{\left(\hat{\mathbf{v}}_{ji}^T {\mathbf{D}_j}^{-1} \hat{\mathbf{v}}_{ji}\right)}} \tag{3.7}$$

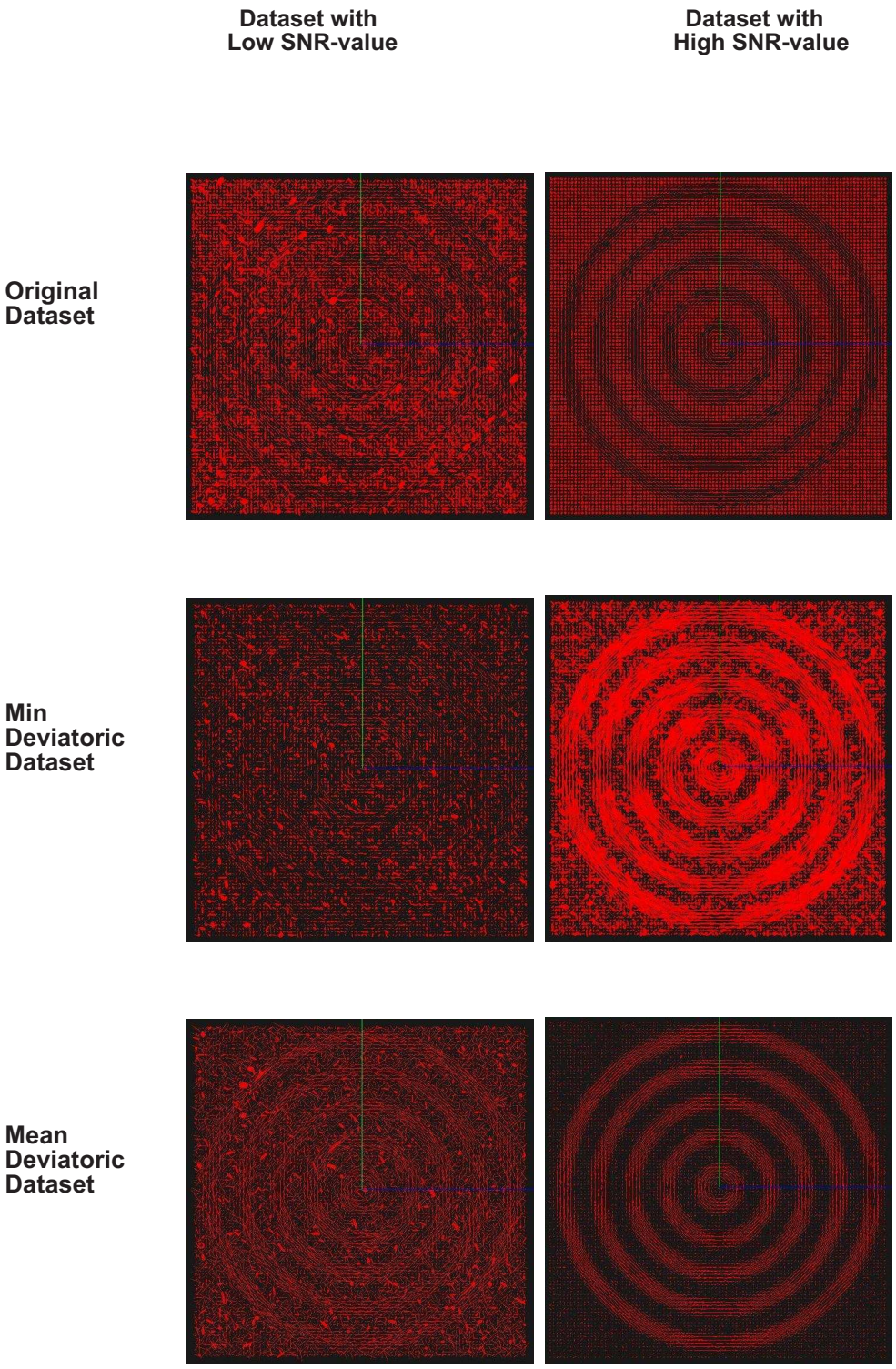

Figure 3.12. Phantom datasets

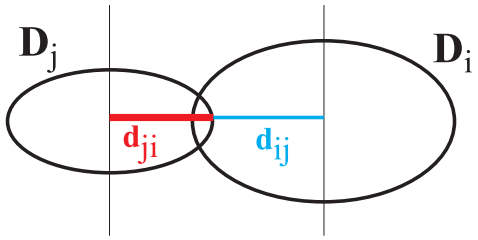

Figure 3.13. Geometrical distance for a Tensor

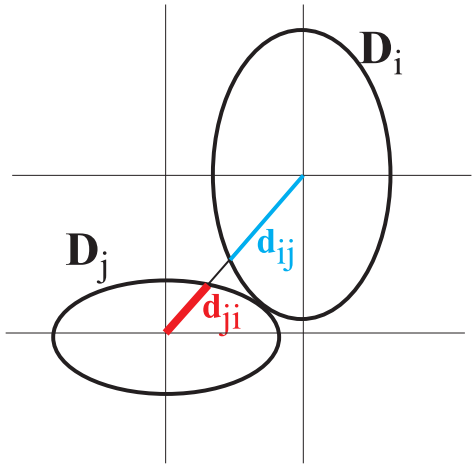

Figure 3.14. Exceptional Case in Geometrical Connectivity

Two tensors are defined to be connected if $d_{ij}+d_{ji} \geq \| \mathbf{r}_i - \mathbf{r}_j \|$. d_{ji}'s are connected for 6-neighborhood scheme.

3.4.2. Probabilistic Connectivity

In the geometric connectivity method above, the 6-neighborhood scheme is used because identifying whether 2 diagonally neighboring ellipsoids touch each other or not, by using d_{ji} is prone to errors. Figure 3.14 demonstrates a case where the ellipsoids touch while $d_{ij} + d_{ji} < \| \mathbf{r}_i - \mathbf{r}_j \|$.

To overcome this drawback of geometric connectivity method, we used the probability of the distribution of diffusing particles(as given by $\tilde{f}(.)$, Eqn 3.2)originating from $\mathbf{r}_j$ to be at the neighboring position i, $\mathbf{r}_i$. This is given by,

$$\tilde{f}(\mathbf{v}_{ji}) = \frac{1}{\sqrt{(4\pi)^3 |\mathbf{D}_j|}} exp \frac{\left(-\hat{\mathbf{v}}_{ji}^T \mathbf{D}_j{}^{-1} \hat{\mathbf{v}}_{ji}\right) {d_{ji}}^2}{4} \tag{3.8}$$

To achieve inherent normalization of with respect to the total diffusion (i.e. to $|\mathbf{D}_j|$) we applied a threshold to the exponential term in $\tilde{f}(.)$ only. This correspond to setting the possibility threshold with respect to the probability of no diffusion(i.e. $\tilde{f}(\underline{0})$). The threshold is set to be [0,1].

This approach considers $\mathbf{D}_j$ only to determine the connectivity between $\mathbf{r}_j$ and $\mathbf{r}_i$. A straight forward improvement of this method is to use a composition of $\mathbf{D}_j$ and $\mathbf{D}_i$ instead. We have used $\mathbf{D}_T = \frac{\mathbf{D}_j + \mathbf{D}_i}{2}$ for an improved version of probabilities corresponding method.

3.4.3. Distance Scaled Mutual Diffusion

In this method, a new connectivity criterion is calculated called "distance scaled mutual diffusion coefficient" for each 26 neighboring tensor pair [24].

$$K_{ji} = \frac{\left[\left(\mathbf{v}_{ji}{}^T \mathbf{D}_j{}^{-1} \mathbf{v}_{ji}\right) * \left(\mathbf{v}_{ij}{}^T \mathbf{D}_i{}^{-1} \mathbf{v}_{ij}\right)\right]^{\gamma}}{\sigma^2} \tag{3.9}$$

where $\hat{v}_{ji}^T D \hat{v}_{ji}$ is the linear diffusion coefficient along $\hat{v}_{ji}$, $\sigma = \| \mathbf{r}_i - \mathbf{r}_j \|$ and γ is empirically set to be 10. $\mathbf{r}_i$ and $\mathbf{r}_j$ are defined to connected if K_{ji} in range [0,20] for original data, K_{ji} in range [0,1] for Min Deviatorric Dataset and K_{ji} in range [0,00001] for Mean Deviatoric Dataset. The reader is referred to [29] for further details referring the distance scaled mutual diffusion coefficient.

3.5. Flow Path Visualization Approach

The work undertaken here as intended to look into the possibility of using techniques developed for flow visualization. For this purpose, a widely used techniques has been selected and applied to diffusion tensor field data.

In the method, the principle eigenvector field of the input diffusion tensor field

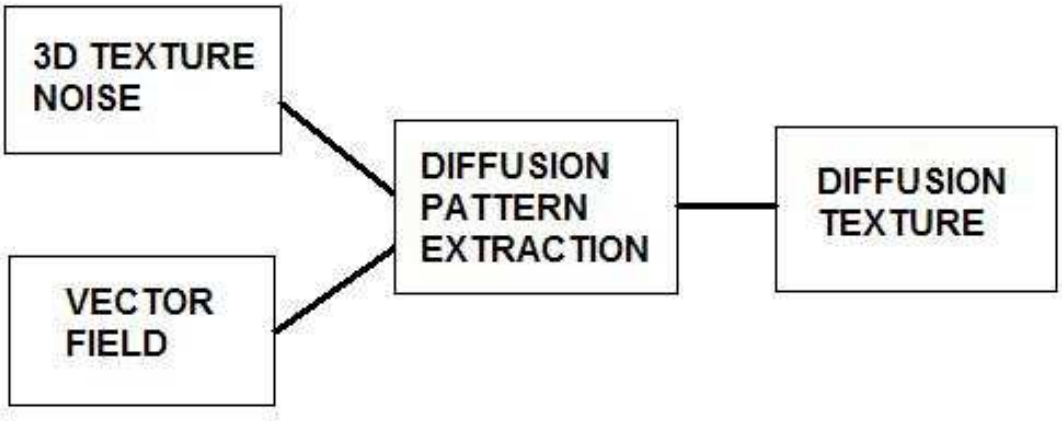

Figure 3.15. Idea behind the Vector Based Concept

is used to modulate an input 3D noise texture with the help of a filter. The output is 3D noise texture which reflects the diffusion pattern(Figure 3.16). The filter we have employed is LIC(Line Integral Convolution).

3.5.1. Line Integral Convolution(LIC)

3-D LIC is a well-known vector field visualization method. [16] In LIC method, there are two basic steps: first one is the local streamline generation and the second is the convolution. For each voxel, a local streamline is computed. Streamline computation in LIC method is basically a Euler integration. Each streamline starts at the center of voxel and moves out in positive and negative directions. Step size can be fixed or dependent on the voxel size. In our method, it is determined by the voxel size. For each step, a local line is computed based on the vector field value of the current voxel point. A streamline consist of such local lines. (Figure 3.17)

At convolution stage, voxels that sit on the local streamlines are used. For each streamline, the volumetric noise values corresponding to the voxels that sits on that streamline are convolved using a weight function.

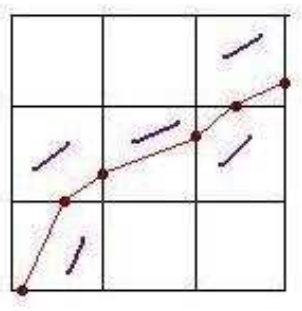

Figure 3.16. Stream Line Generation in 2D

F($\lfloor P_i \rfloor$)is the input voxel corresponding to the vector at position ($\lfloor P_x \rfloor$, $\lfloor P_y \rfloor$, $\lfloor P_z \rfloor$) l is convolution length and h is e weight function in forward direction. l' and h' are same parameters for backward direction.

We have used a gaussian as the weight function. Convolution length in lic method have an effect on the output texture. If convolution length increases, effect on the noise texture will increase. In Figure 3.18, the output textures that are generated for five different convolution lengths are shown.

$$F'(x,y) = \frac{\sum_{i=0}^{l} F(\lfloor P_i \rfloor) h_i + \sum_{i=0}^{l'} F(\lfloor P'_i \rfloor) h'_i}{\sum_{i=0}^{l} h_i + \sum_{i=0}^{l'} h'_i} \tag{3.10}$$

If the resolution of vector field is low, lic may not produce good visualization. In our thesis, to improve the quality of visual understanding,we have increased the resolution by using linear interpolation. Results are shown in Figure 3.19.

When visualizing the data, local streamlines are shown along with the texture. Two FA(fractional anisotropy) limits are used as input parameters for interaction. If FA value of tensor in a voxel is bigger than first fa value, a local streamline that is generated for this voxel is drawn opaque and local lines are drawn in original size. If fa value of tensor is between two fa value, local streamline drawn with a transparency by proportional to the fa value. If fa value of tensor is smaller than second fa parameter, an ellipsoid is used for this voxel. Figure 3.20.

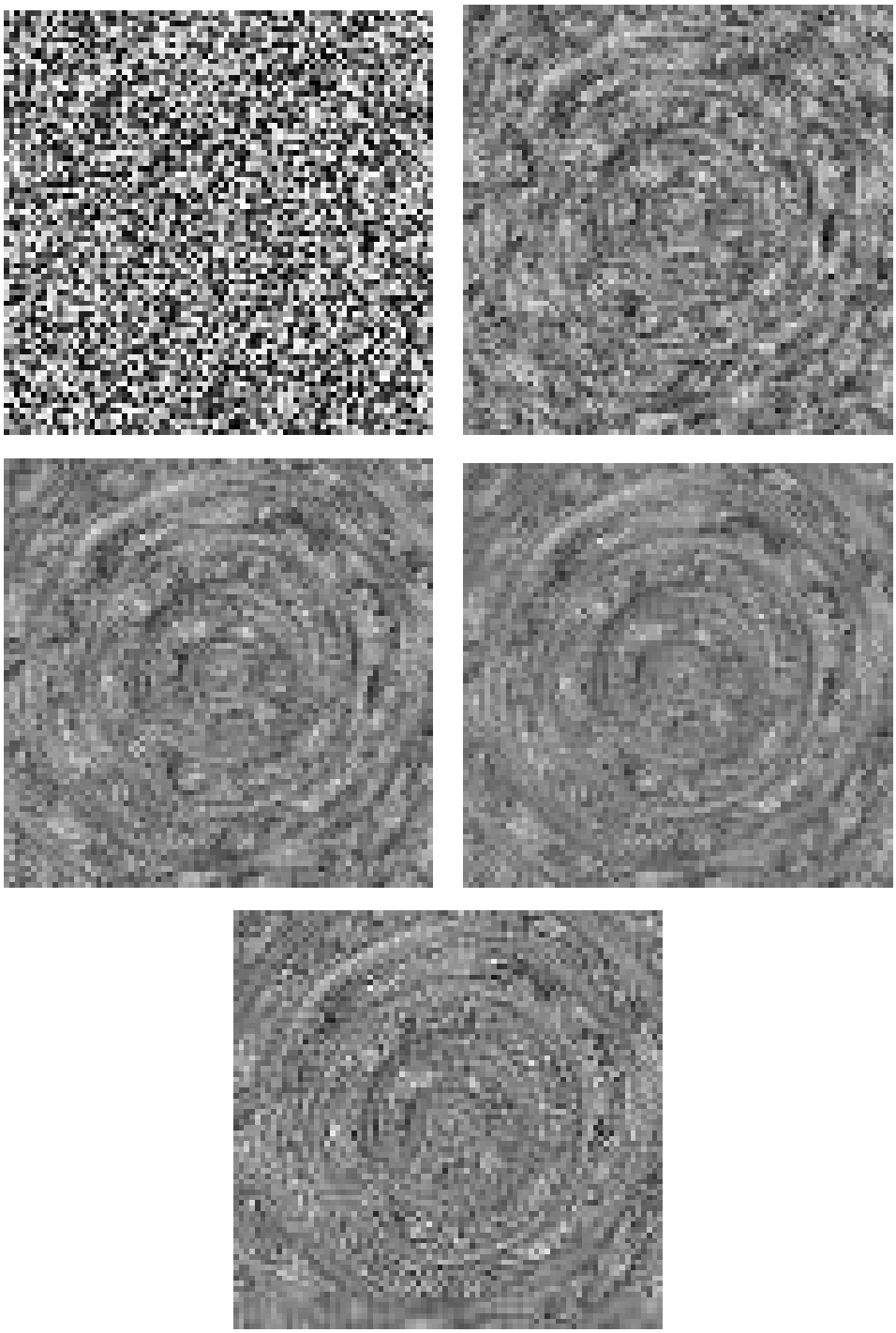

Figure 3.17. Output textures of LIC method on High SNR dataset with convolution lengths: 0, 3, 7, 13, and 17,respectively.

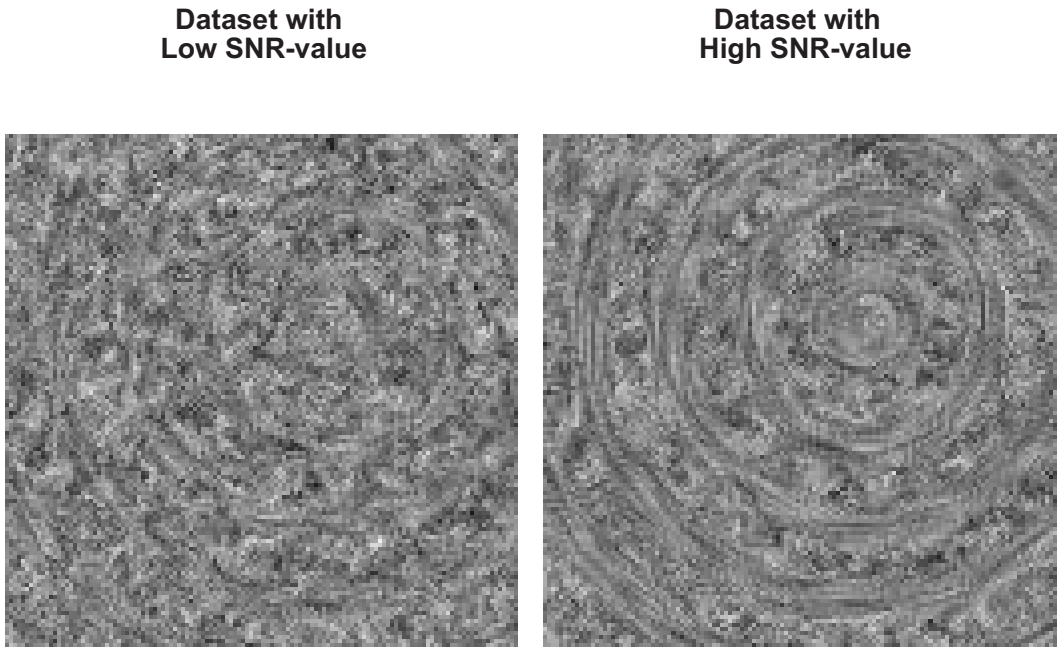

Figure 3.18. LIC Output with interpolated data

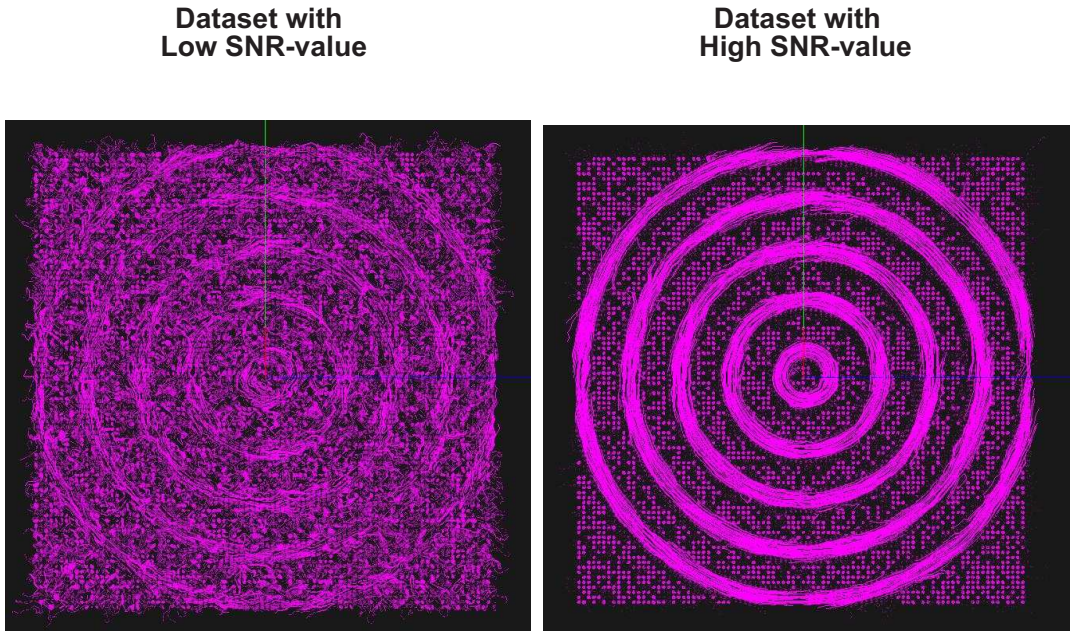

Figure 3.19. Streamlines with original data

3.6. The User Interface of DTVT

The techniques developed for the thesis are integrated into DTVT. For the user interface , Qt[30] programming is used. An OpenGL [31] window is integrated to Qt page to visualize the tensor field. All algorithms are programmed in C++[32]. Results on a phantom data are used to explain the types of user interface components and their functionality.

Three types of visualization methods are integrated to DTVT. Each one is displayed by using its predefined keyboard value. By using **B**, **T** and **L**, *Basic Visualization*, *Tensor Painting* and *LIC* methods will be visible or invisible in the display window, respectively.

3.6.1. Basic Visualization Tool:

In Basic Visualization method, each diffusion tensor is represented by an ellipsoid. Ellipsoids are scaled and oriented according to eigenvalues and eigenvectors of diffusion tensor. Two types of coloring methods are applied. And opacity value of ellipsoids are determined by the Fractional Anisotropy (FA) value of the tensor.

- **C**(Keyboard) *Change Coloring Method* : There are two types of coloring methods. In the first method PDD of each diffusion tensor is used to color ellipsoids. And in the second method, anisotropy coefficients (c_l, c_p and c_s) of diffusion tensor are used. To switch between coloring methods, keyboard *C* is used. Figure 3.20 and Figure 3.21 are examples of first and second coloring methods, respectively.
- **Slice(z)**(Slider) *Change Slice* : Data is displayed slice by slice in the graphical window. By using this slider, tensors that have the selected z-value can be displayed. In the Figure 3.22 and Figure 3.23, two different slices are displayed by using Basic Visualization method.
- **Slice Range**(Slider) *Change Slice Range*: More than one slice can be shown in the screen. Slice range determines the number of slices under and below the selected slice. In Figure 3.24, data is shown in the range of [Slice - *slice range*,

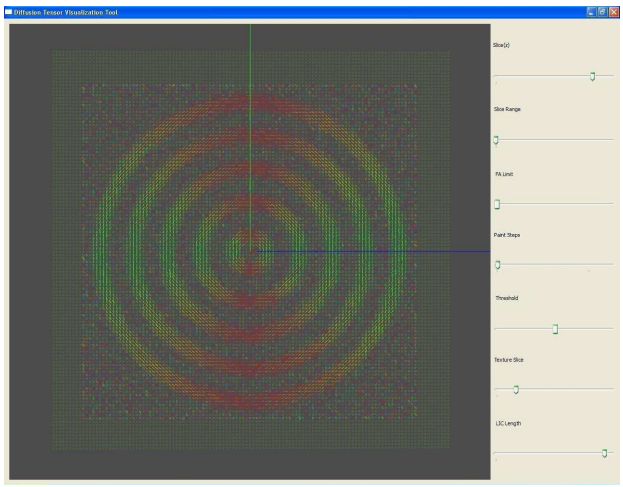

Figure 3.20. Example for first coloring method that uses the PDD of each tensor (Use of Keyboard ”C”)

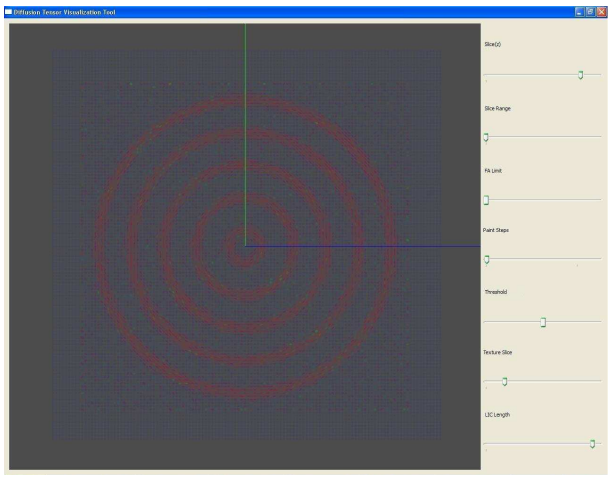

Figure 3.21. Example for first coloring method that uses the Anisotropy Coefficients of each tensor (Use of Keyboard “C”)

Slice + *slice range*].

- **FA Limit**(Slider) *Change FA(Fractional Anisotropy) limit*: A limit is defined for FA value of tensors. Only tensors that has FA value higher than this limit are shown. In Figure 3.25 and Figure 3.26, Tensor field is visualized by selecting two different FA threshold.

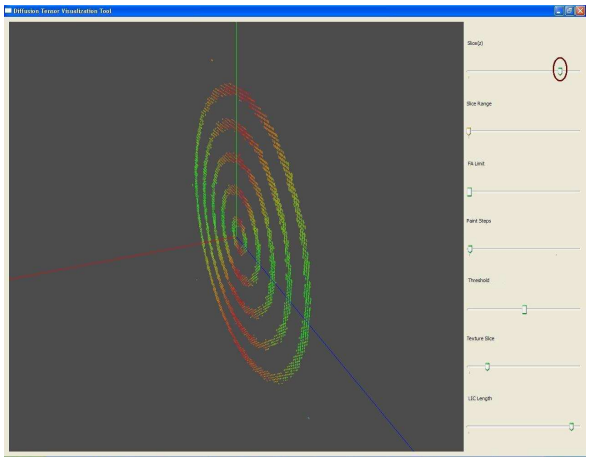

Figure 3.22. Seventh slice of Diffusion Tensor Field is displayed (Use of “Slice (z)” Slider)

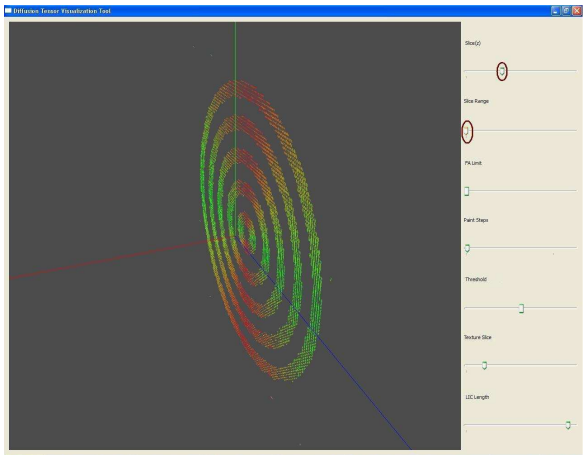

Figure 3.23. Fourth slice of Diffusion Tensor Field is displayed (Use of “Slice (z)” Slider)

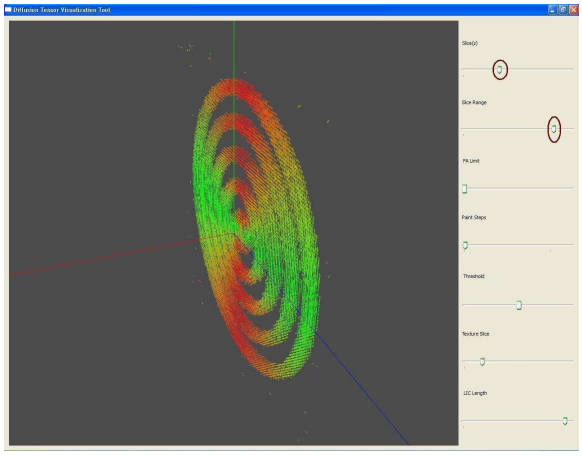

Figure 3.24. Slices of Diffusion Tensor Field is displayed in the range[2-7] (Use of “Slice Range” Slider)

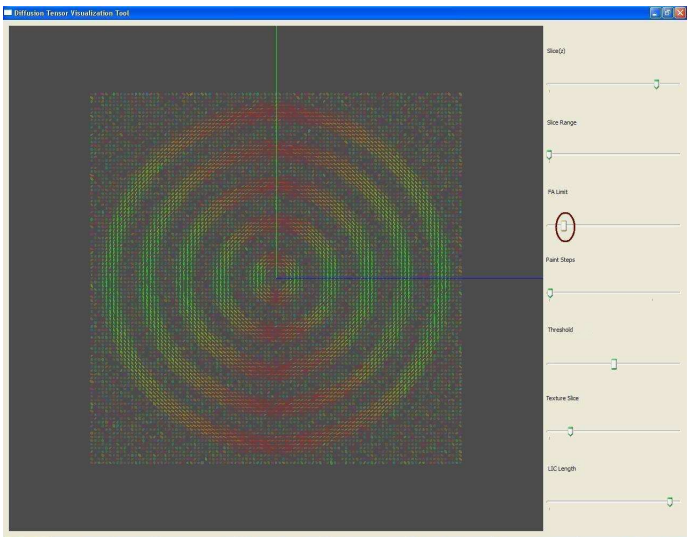

Figure 3.25. All tensors in the selected slice are displayed (Use of "FA Limit" Slider)

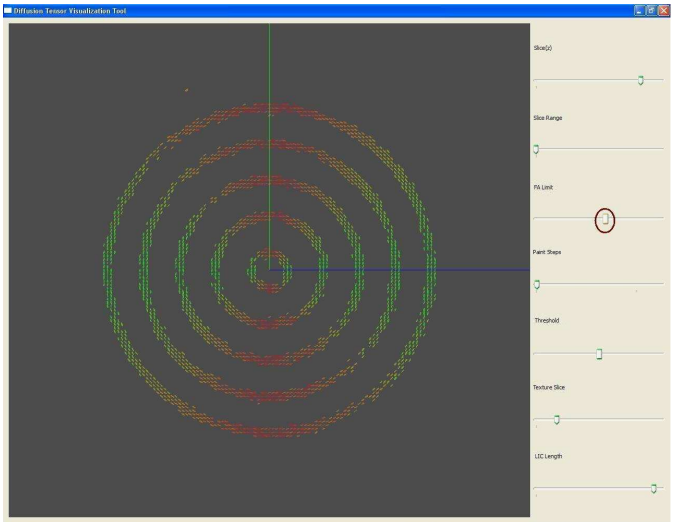

Figure 3.26. Only tensors whose fa value is higher than FA limit are displayed (Use of "FA Limit" Slider)

3.6.2. Tensor Painting Tool:

In Tensor Painter method, glyphs, that define diffusion tensors, are painted gradually as growing regions by using the connectivity between tensor pairs. By spreading paint in the tensor field, the diffusion is visualized. Four types of connectivity measures are used. And methods are applied to three datasets.We used three main parameters for interaction with interface.

- **Paint Step**(Slider) *Change Painting Step*: In tensor painting technique, paint diffuse from seed tensors to other neighboring tensors step by step. In each step, tensors that are painted in the last pass, paint their neighbors. By showing painted tensors step by step, we can see the flow of paint in tensor field in 3D.(Figure 3.27)
- **FA Limit**(Slider) *Change FA Limit*: For Tensor Painting Method, FA limit determines the leakage criteria. Tensor are painted in color white if they are painted before leakage occurs, otherwise they are painted in color red. In Figure 3.28, from the top image to bottom image, only the FA Limit value decreases while the threshold value is same. If the FA limit decreases, paint leaks later and more tensors are painted in color white.
- **Threshold**(Slider) *Change the Threshold of connectivity constants*: By changing the other decision criteria, connectivity threshold, motion of paint can be controlled. In Figure 3.29-32, from the first image to fourth image, threshold value increases while FA limit is same. In the first image, since the threshold is low, paint flows easily and lots of tensor are painted but it also leaks easily and only some of them painted in color white. When the threshold increase, the number of painted tensors decrease but leakage occurs later and more tensor painted in color white. In the fourth image, the number of tensors that are painted before leakage occurs decreases since paint cannot paint them because of the high threshold. As we can see, there is an optimum threshold value to paint more tensor before leakage.
- + / - (Keyboards) *Increase or decrease the medium of threshold value in the Threshold Slider.*

In tensor painting method, three datasets are used. These are *Original* and two sets of preprocessed data, *Min Deviatoric* and *Mean Deviatoric*. For each painting method, a connectivity map is calculated by using the selected dataset. DTVT creates *ConnectivityMaps.txt*, *ConnectivityMaps0.txt* and *ConnectivityMaps1.txt* files for Original, Min Deviatoric and Mean Deviatoric datasets, respectively. Each file includes connectivity maps of four painting methods. If DTVT can find these files, it reads them, Otherwise it creates them.

To select the dataset and painting method, these keyboard values are used.

- **Q** (Keyboards) *Loads Original Dataset.*
- **A** (Keyboards) *Loads Min Deviatoric Dataset.*
- **Z** (Keyboards) *Loads Mean Deviatoric Dataset.*
- **1 / 2 / 3 / 4** (Keyboards) *Change painting method.* Select the method number for Tensor Painting.

3.6.3. LIC Texture Tool:

In LIC method, 3D noise texture is modulated by using the local streamlines for each voxel. We have used a number of parameters to change the property of the texture.

- **I** (Keyboards) *Makes Streamlines Visible/Invisible* : By using Keyboard I function, we can display LIC texture with/without streamlines that are generated in LIC method.
- **LIC Length**(Slider) *Change Convolution Length*: In the application of LIC(Line Integral Convolution) method to 3D tensor data, Convolution Length determines the how noise is smoothed by using texture. Figure 3.33-35, three LIC texture (different convolution lengths) are shown with/without their streamlines
- **Texture Slice**(Slider) *Change Texture Slice*: Texture is displayed for each slices. By using this slides, slice number of the texture can change. In the Figure 3.36 and Figure 3.37, two different slice are displayed.

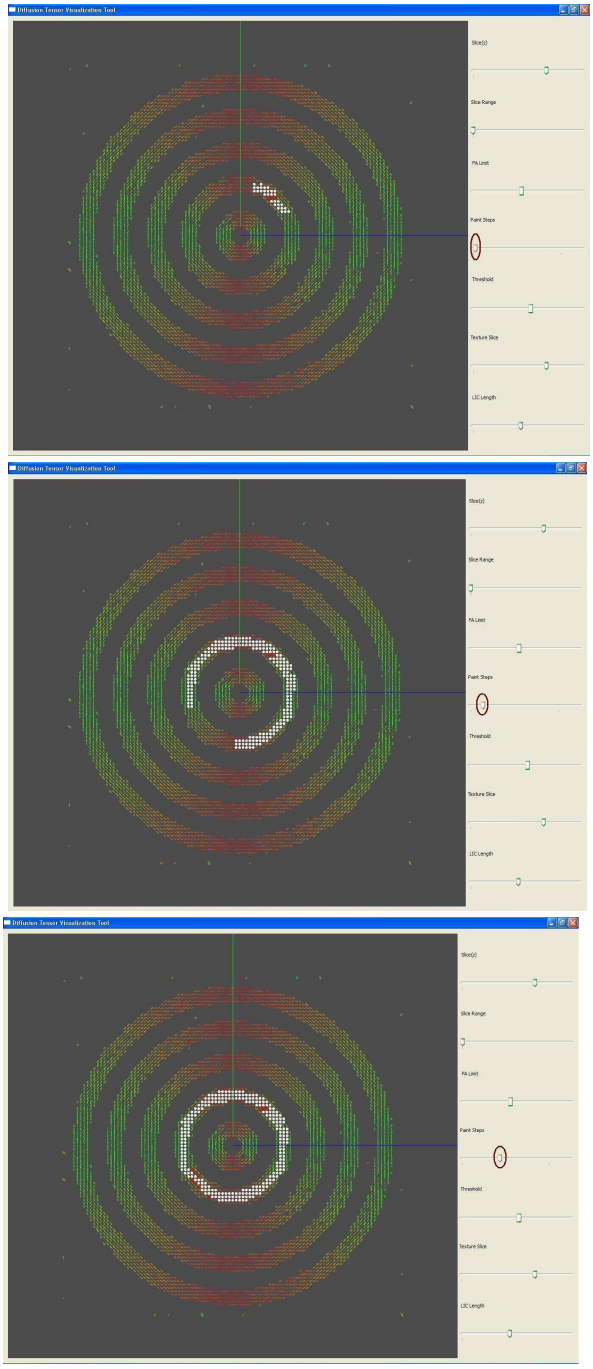

Figure 3.27. Tensors are painted in stepwise fashion by incrasing and decreasing step value(Use of “Paint Steps” Slider). Note: Painting started from the middle of the top right quarter of the second ring.

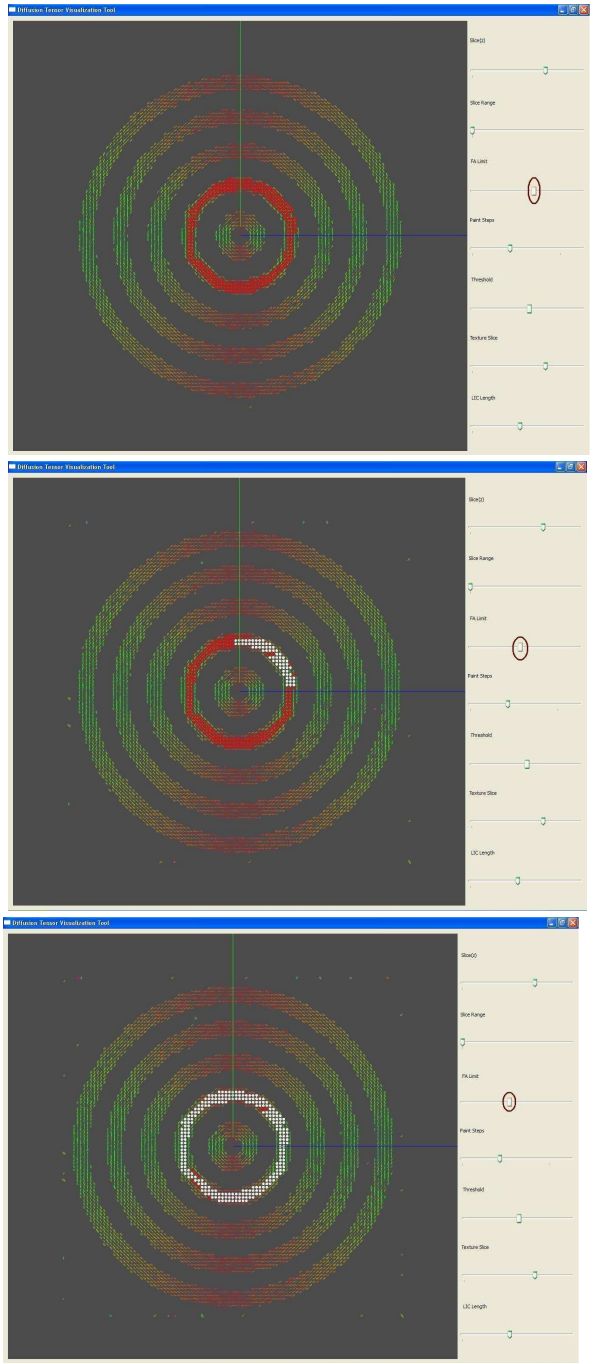

Figure 3.28. Tensor are painted in color white if they are painted before leakage occurs, otherwise they are painted in color red.(Use of “FA Limit” Slider). Note: Painting started from the middle of the top right quarter of the second ring.

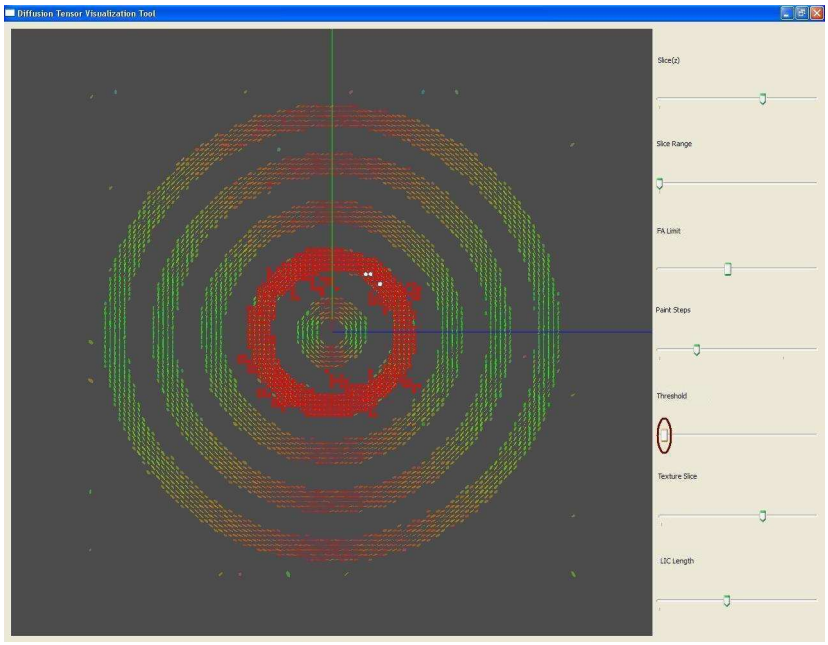

Figure 3.29. Tensors are painted by changing threshold value(Use of "Threshold" Slider). Note: Painting started from the middle of the top right quarter of the second ring.

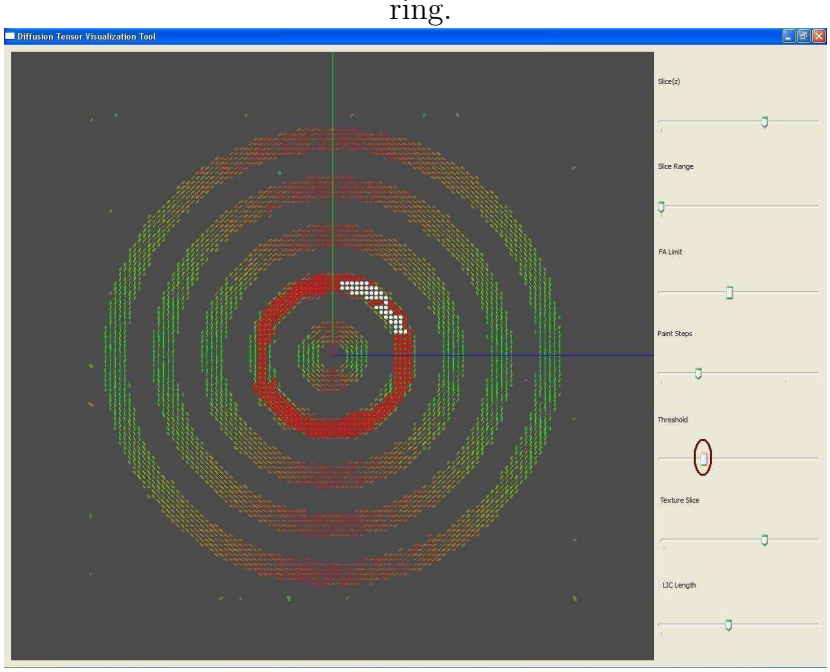

Figure 3.30. Tensors are painted by changing threshold value(Use of "Threshold" Slider).

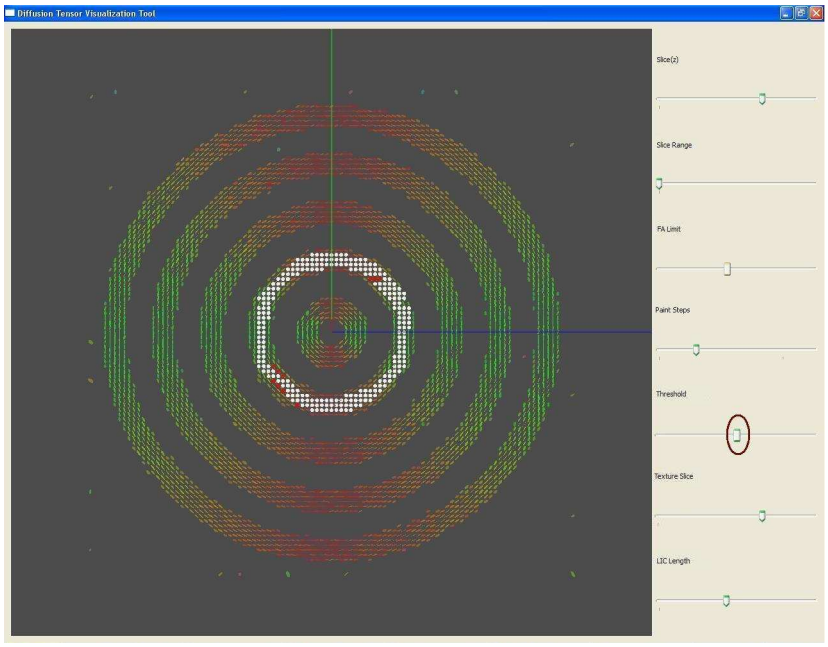

Figure 3.31. Tensors are painted by changing threshold value(Use of "Threshold" Slider).

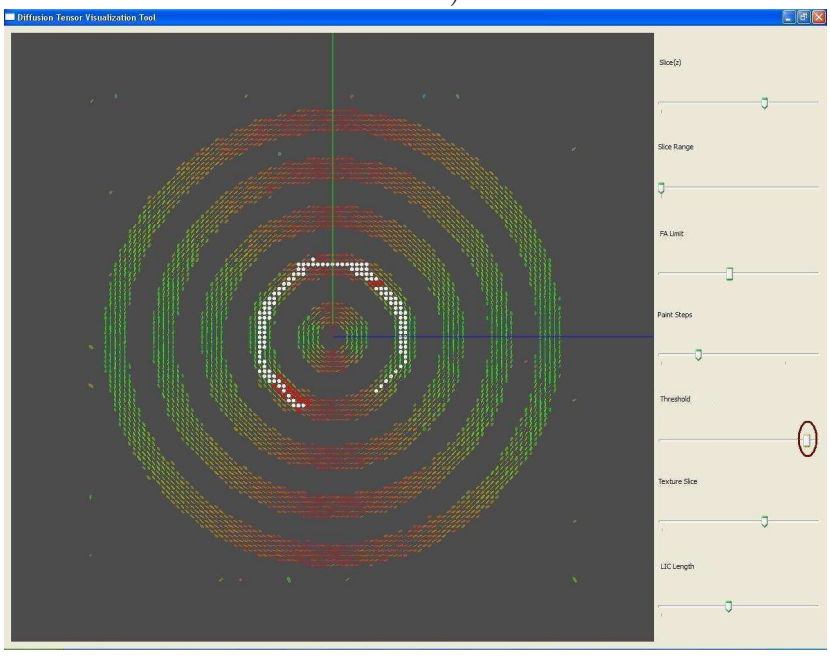

Figure 3.32. Tensors are painted by changing threshold value(Use of "Threshold" Slider).

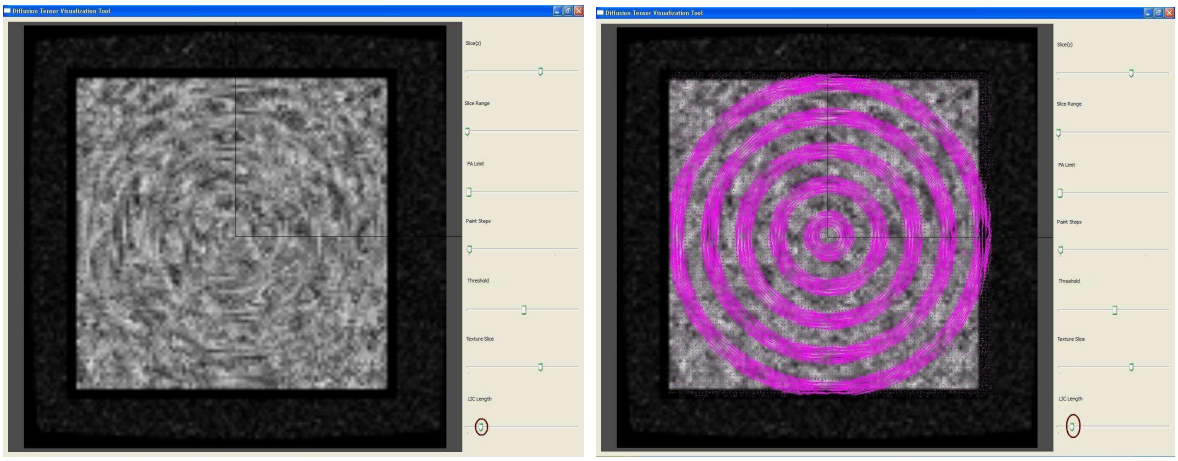

Figure 3.33. The LIC image with low Convolution Length and Stream Lines that are generated for LIC Method (Use of “LIC Length” Slider)

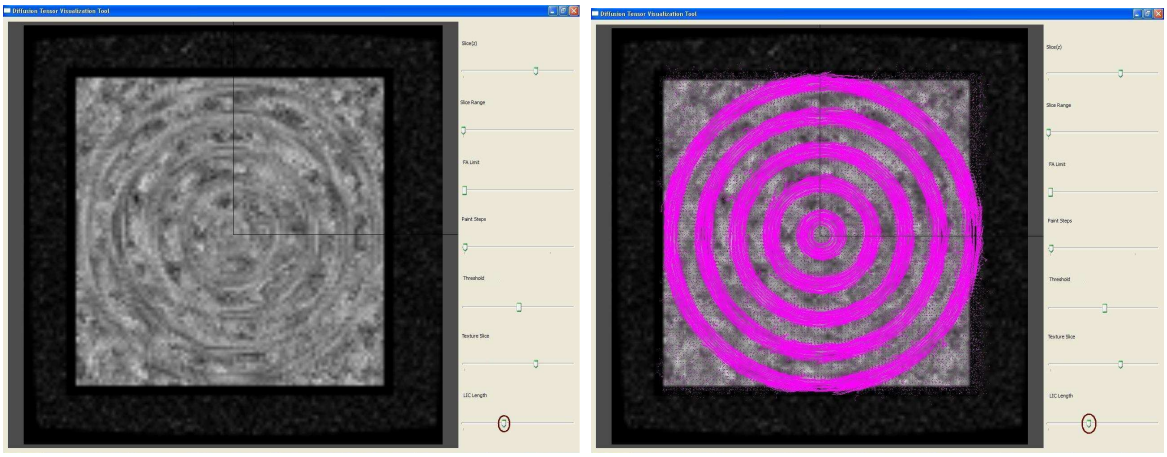

Figure 3.34. The LIC image with medium Convolution Length and Stream Lines that are generated for LIC Method (Use of “LIC Length” Slider)

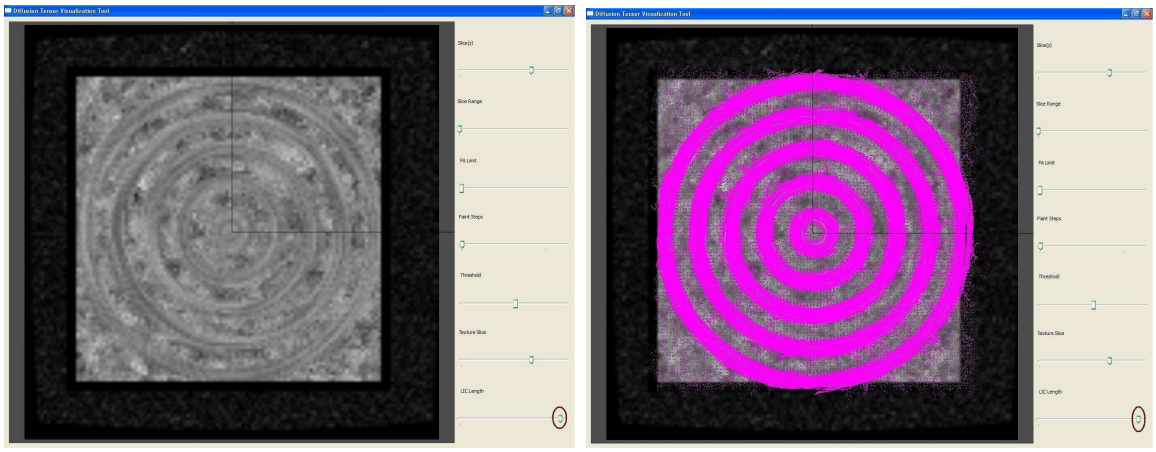

Figure 3.35. The LIC image with high Convolution Length and Stream Lines that are generated for LIC Method (Use of “LIC Length” Slider)

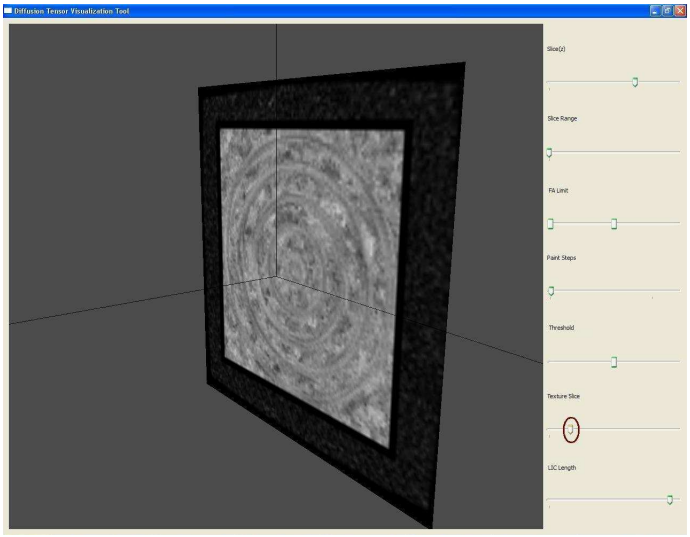

Figure 3.36. LIC image of Slice 3 (Use of "Texture Slice" Slider)

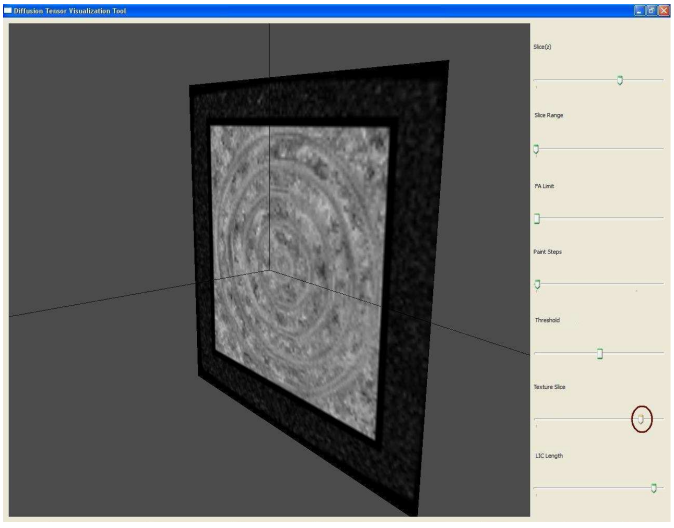

Figure 3.37. LIC image of slice 7 (Use of "Texture Slice" Slider)

4. EVALUATION

Tensor field visualization is a challenging task because of the multi-variate nature of tensor data. Tensor user interfaces are difficult to design, given that human beings are not accustomed to mentally integrate data with so many parameters. Therefore, there are no well-defined and preferred methods in these area. And also, there is no well-known method for evaluation of tensor visualization methods.

To evaluate two new methods proposed in this thesis, two evaluation methods have been developed. For tensor painting, we have tried to measure the path legths that are tracked(painted) without any what we call "leakage" occurs. We compare results by changing the curvature of the diffusion area with different snr-valued datasets. For LIC methods, we tried to find the correlation between the output image of LIC and Fractional Anisotropy map of the same slices. We present the results with varying convolution length parameter evaluation was carried out on datasets with different SNR-value.

This chapter explains the methodology of two evaluation methods presents the results in the form of graphs. They are discussed and some special cases are examined in detail.

4.1. Methodology

Tensor visualization methods are evaluated using phantom 3D tensor fields obtained by simulated DTI data. The DTI simulations are performed on mathematically defined fiber structures, which are in the form of five concentric rings of different curvatures [25]. The goal of the evaluation is to assess the performance of the proposed techniques in visualizing these ring structures using the associated tensor fields. All evaluations are performed twice each with different SNR values.

4.1.1. Tensor Painting

In tensor painting, four method are proposed. We called *Distance Scaled Mutual Diffusion* method as *Method 1*, *Improved Probabilistic Connectivity* method as *Method 2*, *Probabilistic Connectivity* method as *Method 3*, *Geometrical Connectivity Between Diffusion Tensors* method as *Method 4*. For tensor painting, each method is applied to the original data as well as to two sets of preprocessed data. The preprocessing involves calculating the anisotropic(deviatoric) part of the tensors using two separate methods. In the first method, the eigenvalues of each tensor is shifted such that the minimum eigenvalue is 0. This set is called the *Min Deviatoric*. The second method involves shifting the eigenvalues such that their mean value is equal to 0. The negative eigenvalues, created as a result of this shift, are set to 0. This set is called the *Mean Deviatoric*.

The maximum path distance that could be tracked (painted) before there is any leakage outside the fibers (ie. rings)is measured. A leakage is defined to be the case where a tensor with an FA value less than 0.5 is painted. This process is repeated for each ring, for both SNR levels and for a range of local connectivity thresholds. The point plots for maximum tracked distance vs the threshold are computed in each case. We determined the maximum tracked distance over all threshold values used for each case. The results for different preprocessed datasets and tensor painting methods are reported.

4.1.2. LIC

The 3D LIC algorithm was applied to the PDD(Principle Diffusion Direction) vector field. The algorithm takes a 3D white noise and the PDD field as its input and generates a scalar 3D pattern correlated with the local orientation of the PDD field. The evaluation of the LIC method is based on the correlation of the structure of the computed 3D scalar field(the pattern) with the fractional anisotropy(FA) map associated with the input PDD field(ie. with the underlying DT-MRI data). The structure of the LIC output is represented using a scalar map computed as follows:

Let f(r) be the scalar LIC output, then

$$I = [[\nabla f(r)][\nabla f(r)^T]] * H(r) \tag{4.1}$$

is the structure tensor of f(r), where $\nabla f(r)$ is the 3D gradient of f(r), H(r) is a 3×3×3 Moving Average smoothing kernel and '*' denotes the 3D convolution operation. The eigenvalues of I, $\lambda_1 <= \lambda_2 <= \lambda_3$ provide information about the structure of f(r). $\lambda_1 = \lambda_2 = \lambda_3$ corresponds to contrast noisy regions, while $0 = \lambda_1 = \lambda_2 < \lambda_3$ corresponds to a pettern composed of elongated, line-like structure, similar to the pattern we get by LIC for fibers in the DT-MRI data. We used

$$\kappa = \sum_{i=1}^{2} \sum_{j=i+1}^{3} (\lambda_i - \lambda_j)^2 \tag{4.2}$$

as the single parameter representing the structure of f(r). The evaluation is based on the correlation coefficient between the κ-map(κ) and the FA-map(F).This is defined as

$$\rho = \frac{cov(F(r), \kappa(r))}{\sigma_F \sigma_\kappa} = \frac{\frac{1}{N-1} \sum_r (F(r) - \bar{F})(\kappa(r) - \bar{\kappa})}{[\frac{1}{N-1} \sum_r (F(r) - \bar{F})^2]^{\frac{1}{2}} [\frac{1}{N-1} \sum_r (\kappa(r) - \bar{\kappa})^2]^{\frac{1}{2}}} \tag{4.3}$$

where $\bar{F}$ and $\bar{\kappa}$ are mean values, and the summation are defined for the whole volume which is composed of N voxels. ρ is a real number between -1 and 1. The closed $\| \rho \|$ to 1, the higher the correlation.

4.2. Results and Discussion for Tensor Painting

Figure 4.1 - 4.4 presents the bar charts for four different methods. Each one shows the maximum tracked distance(as arclegth) before any leakage occurs, for each dataset pair, for varying fiber ring curvature. Note that each one of these maximum

distances may be obtained with different local connectivity thresholds values. In fact, the variation of these optimum thresholds for each preprocessing and painting method is an indicator of how well we can define an optimum operating point for all curvature values. These variation are reported in Figure 4.5 - 4.8, and we will discuss them later as the best performing methods.

The fourth method does not provide meaningful results for the evaluation criteria. Performance is not dependent on snr value of data and the curvature. Therefore, this method is ignored in discussion.

When we analyze the bar charts(in Figure 4.1-4.4) that shows the maximum tracked distance vs threshold information, we can say that, data with high snr value produce longer maximum tracking without leakage than data with low snr value for each method. Moreover, each dataset have different sensitivity for the change in SNR value. In the Min Deviatoric dataset, influence of snr is enhanced. When the results of unprocessed data are compared, second method gives better results. In the case of processed data, maximum tracking occurs when first method is used.

To analyze the effect of change in curvature, we tried to paint fiber rings with different radii in the diffusion tensor field. The curvature of a ring with radius r is proportional to (1/r). We can observe that when the radius of rings increase, maximum tracked distance also increases. Therefore, we can say that tracking is better, whit low curvature values. Effect of curvature becomes more pronounced when high SNR data is used. The performance gets better with decreasing curvature in high SNR. This is not always the case for low SNR because of noise. It can be observed that these is a slight performance degradation. When the performance for ring with curvature (1/45)unit is compared with the performance for ring with curvature (1/35)unit, this is a unexpected result, it may be due to data generation.

Consequently, we have concluded that using method 1 with Min Deviatoric for high SNR data and using method 1 with Mean Deviatoric for low SNR data seems to be the best choices. These two cases are examined in the Figures 4.9 and 4.10.

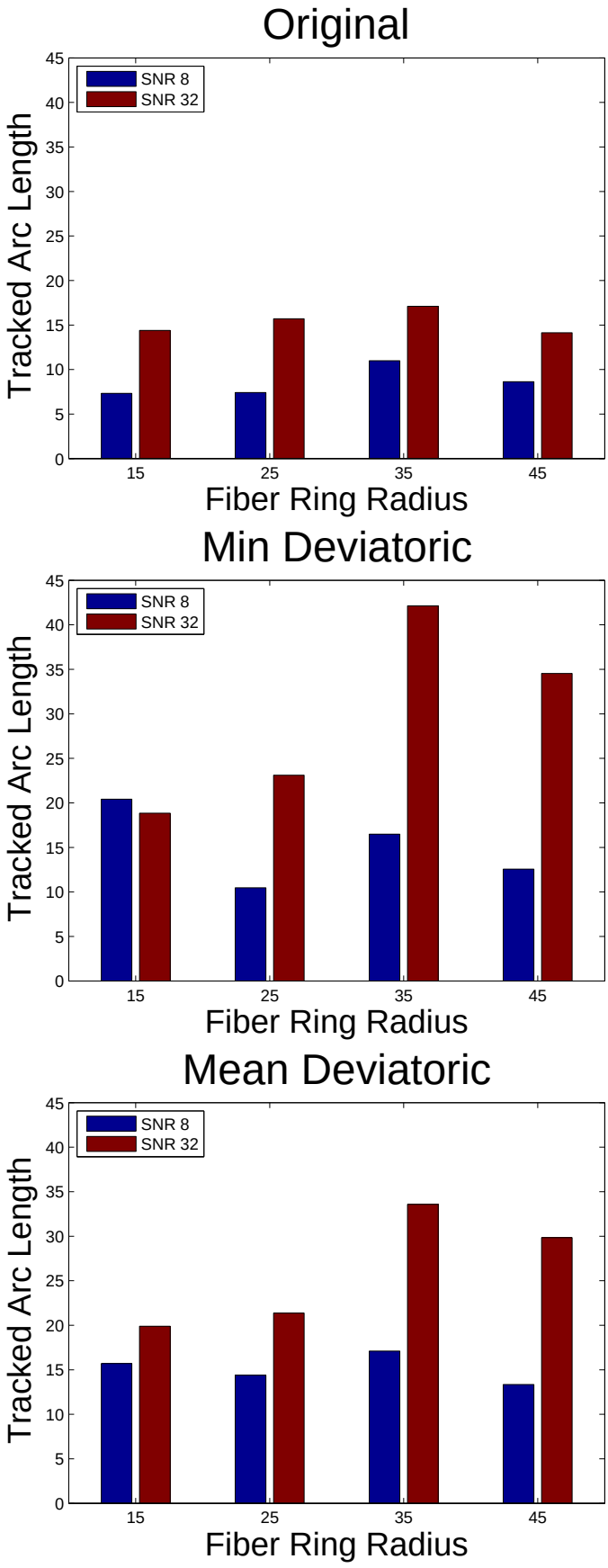

Figure 4.1. Maximum tracked Arc Length without leakage for different Curvature values (Method 1)

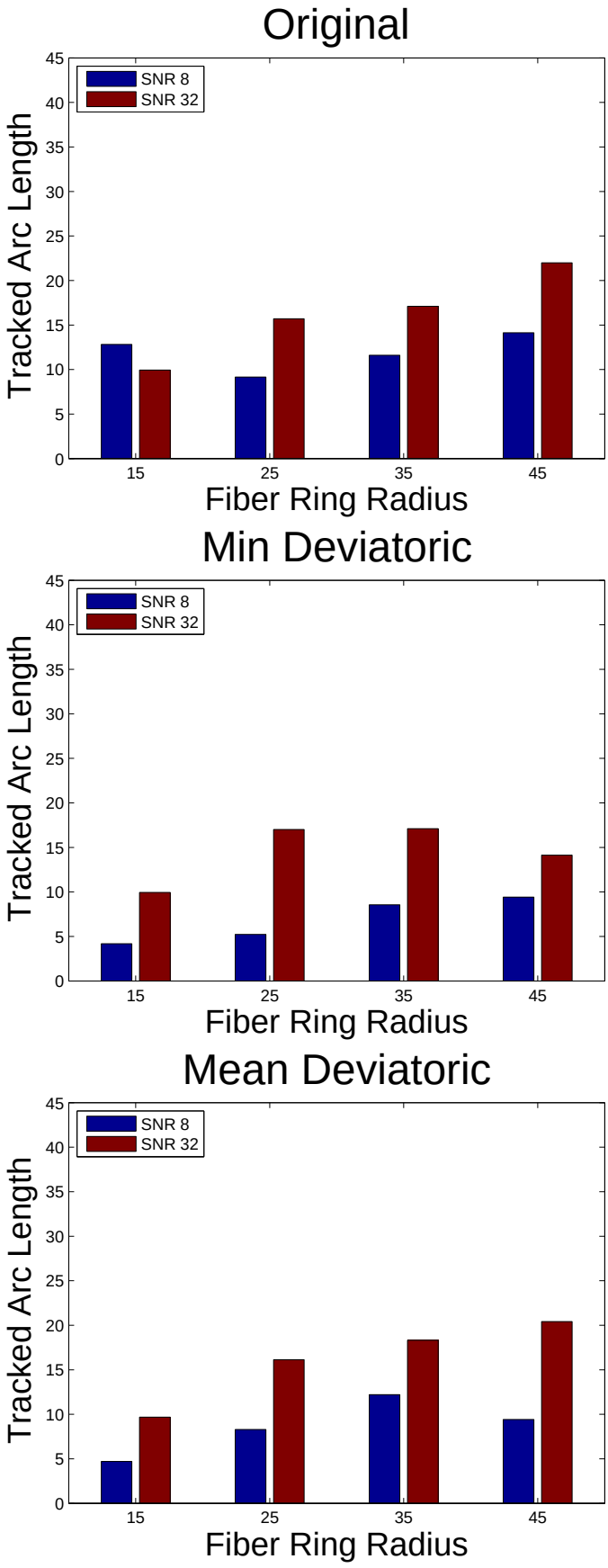

Figure 4.2. Maximum tracked Arc Length without leakage for different Curvature values (Method 2)

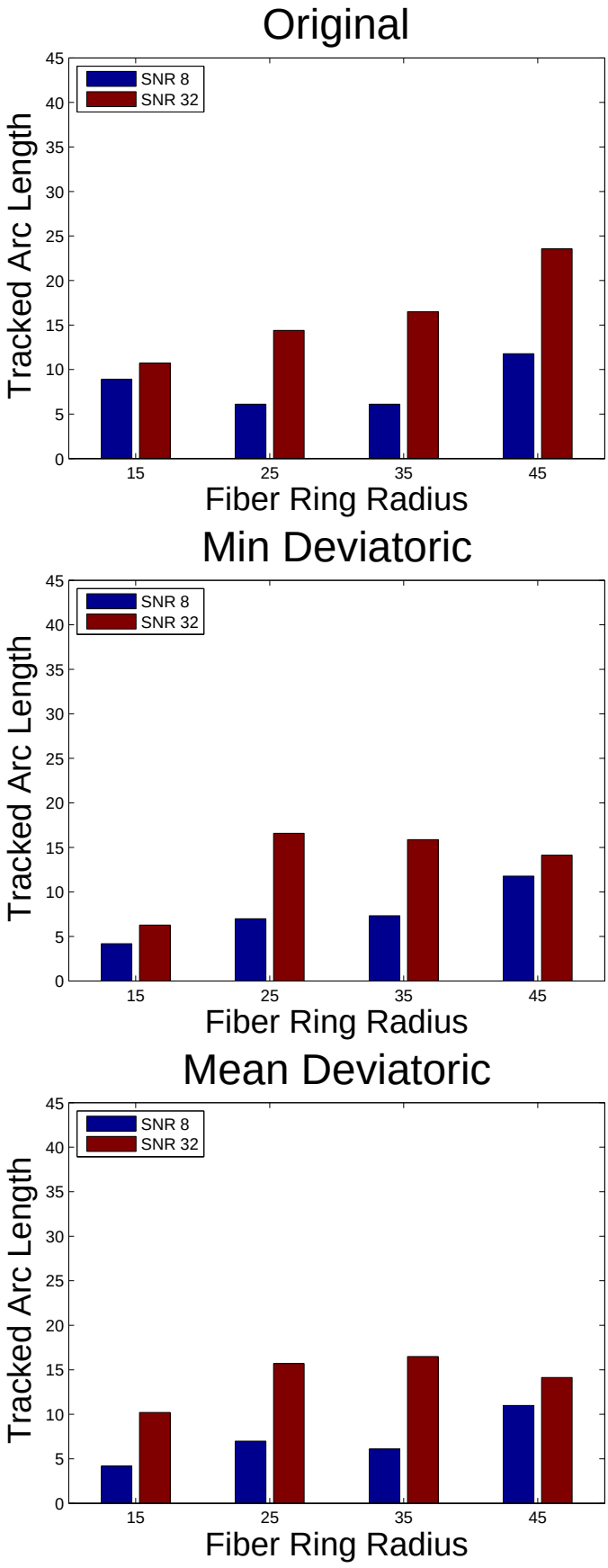

Figure 4.3. Maximum tracked Arc Length without leakage for different Curvature values (Method 3)

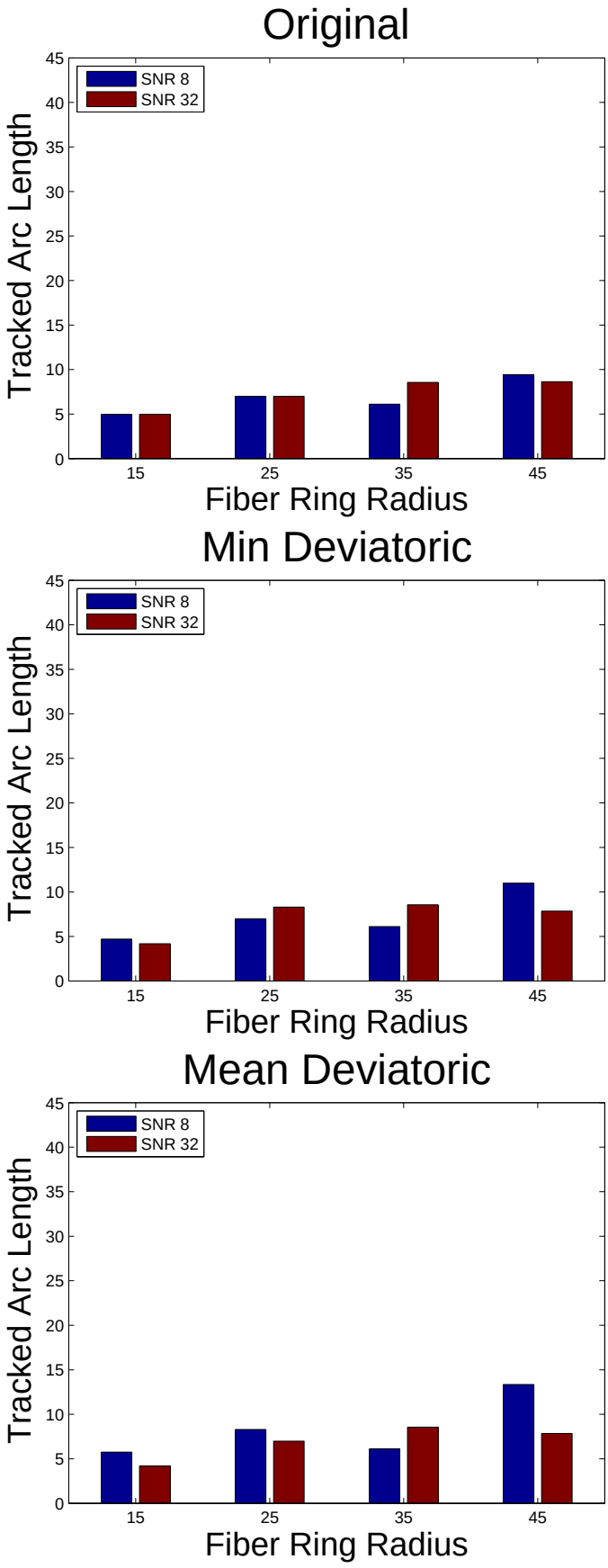

Figure 4.4. Maximum tracked Arc Length without leakage for different Curvature values (Method 4)

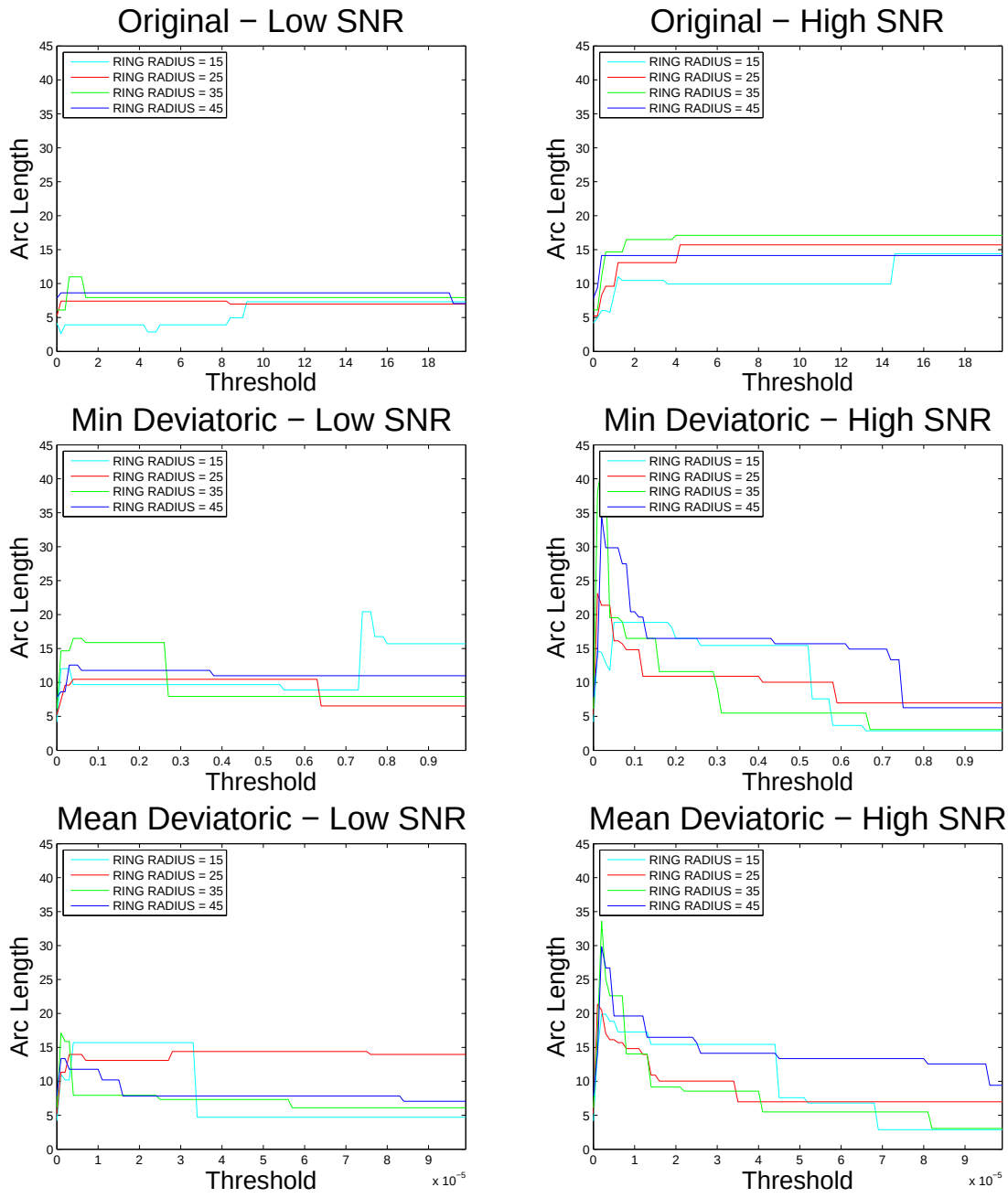

Figure 4.5. For Method 1, Change in distance tracked without leakage for a range of threshold

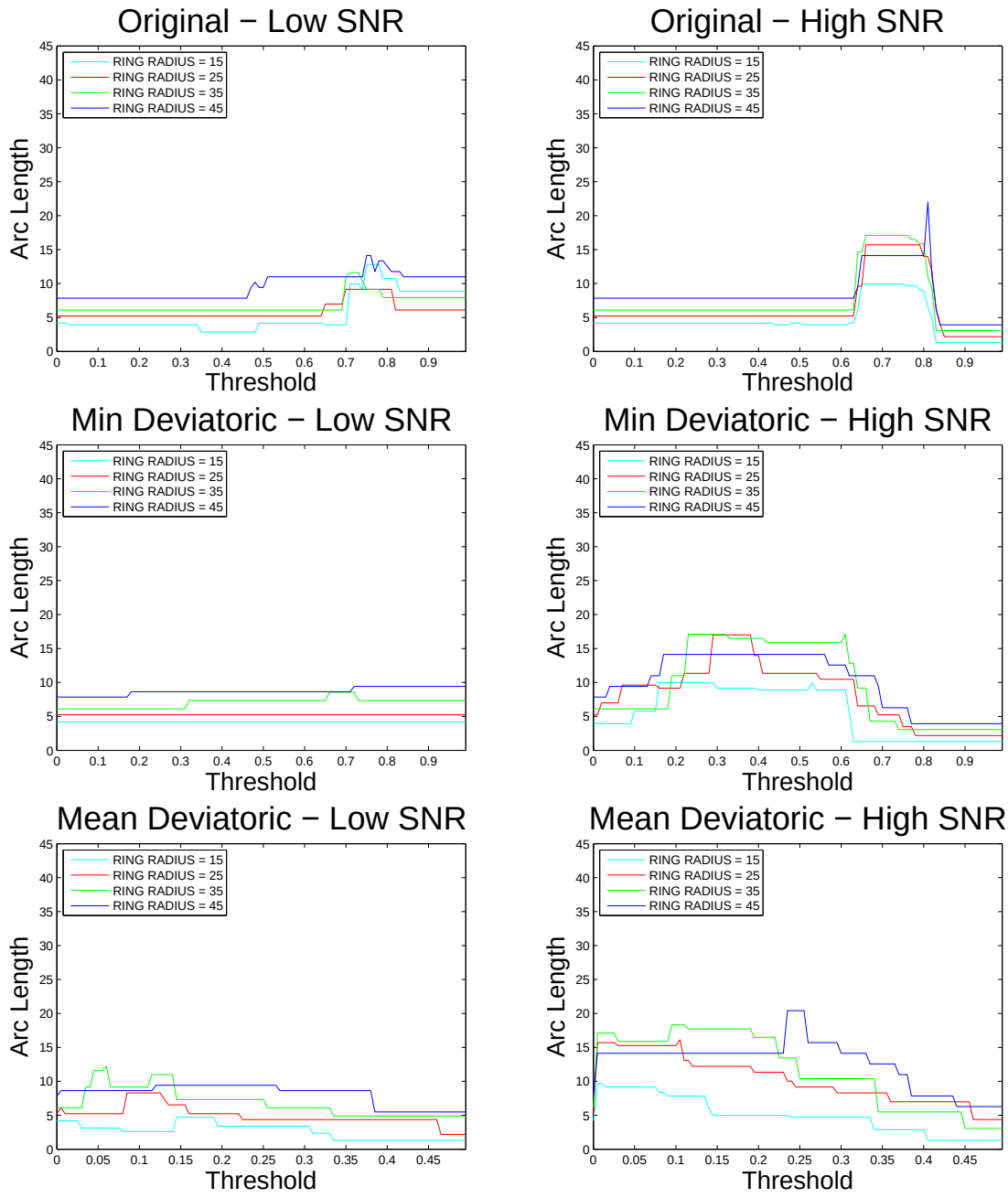

Figure 4.6. For Method 2, Change in distance tracked without leakage for a range of threshold

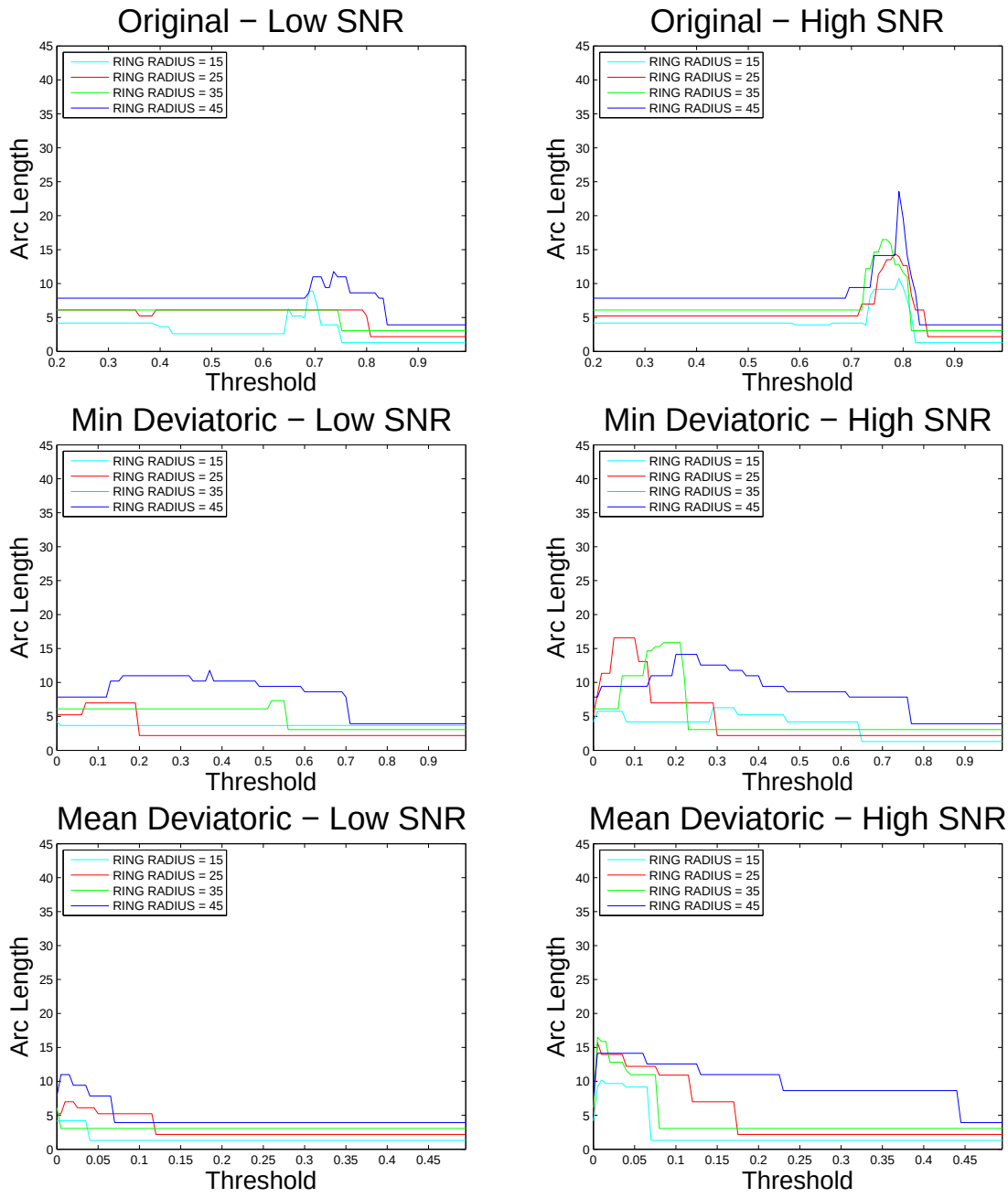

Figure 4.7. For Method 3, Change in distance tracked without leakage for a range of threshold

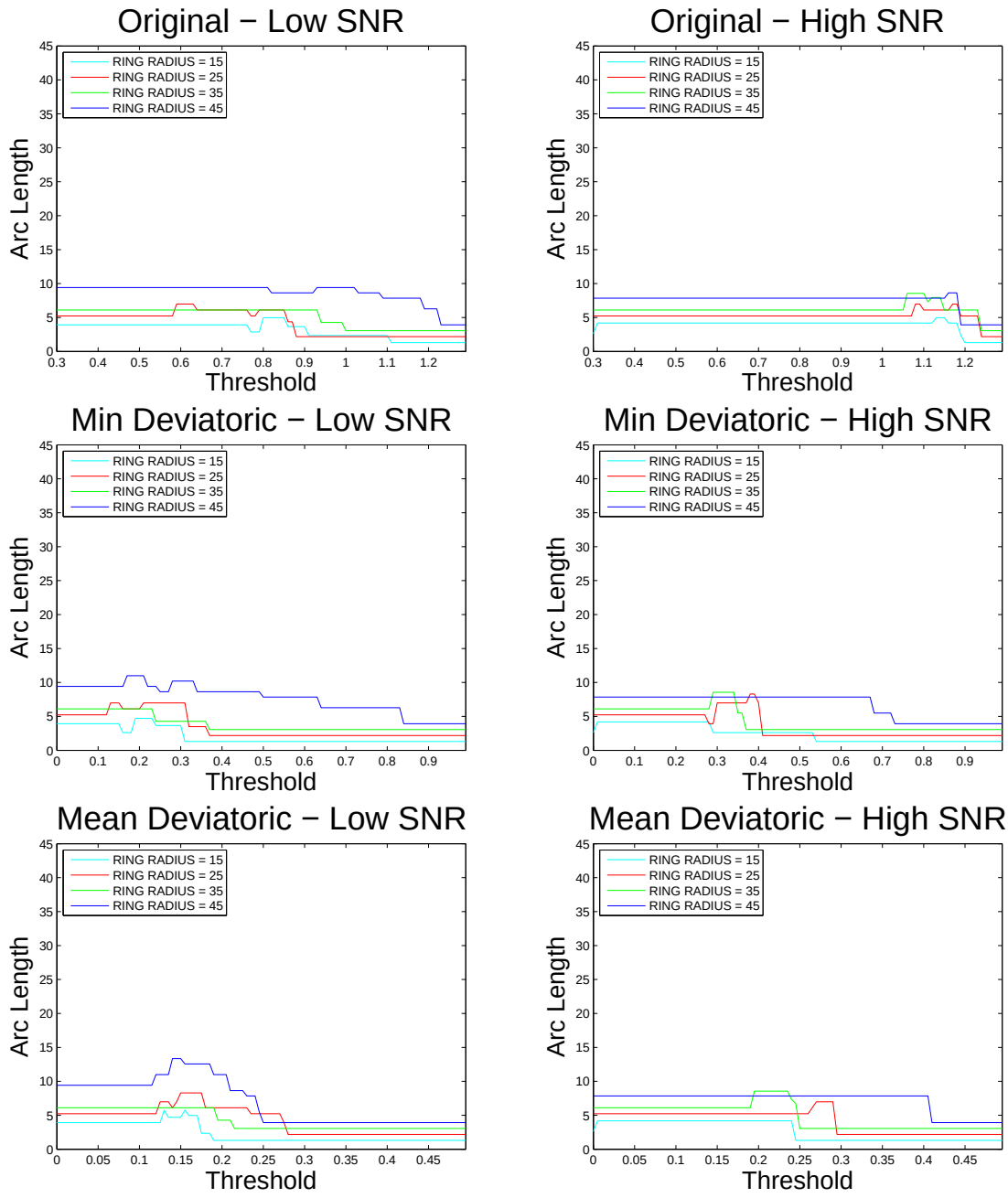

Figure 4.8. For Method 4, Change in distance tracked without leakage for a range of threshold

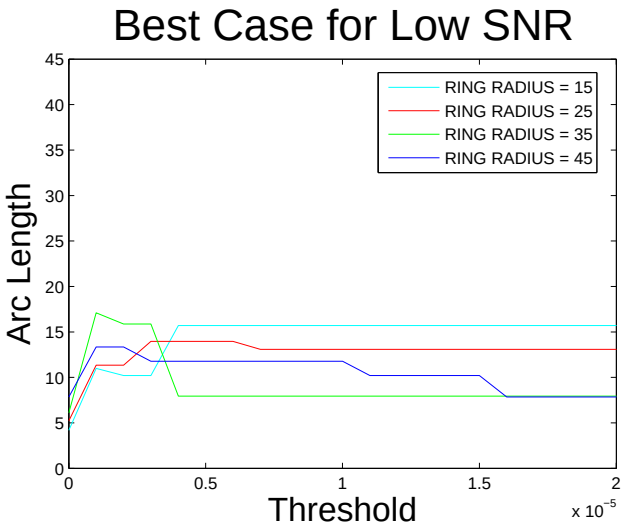

Figure 4.9. Arc Length vs Threshold of Selected Case for Low SNR

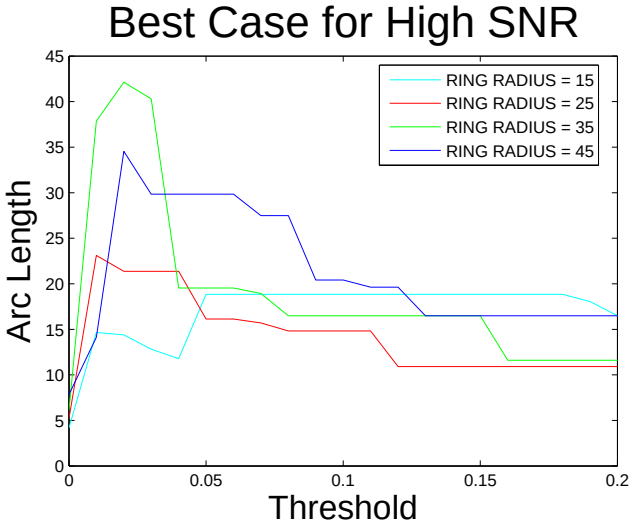

Figure 4.10. Arc Length vs Threshold of Selected Case for High SNR

In these two cases, maximum tracked distance without leakage is higher than the other cases. Moreover, when the arc length vs threshold graphs of these two cases are examined, we can observe that an optimum threshold region for all four rings can be set to obtain the maximum arc length is occurs.

4.3. Results and Discussion for LIC Method

The LIC method generates what we call texture images. Figures 4.11 and 4.12 shows the correlation between the fa map and output texture for low and high SNR-values respectively generated for 7th slice of phantom diffusion tensor field issued. The

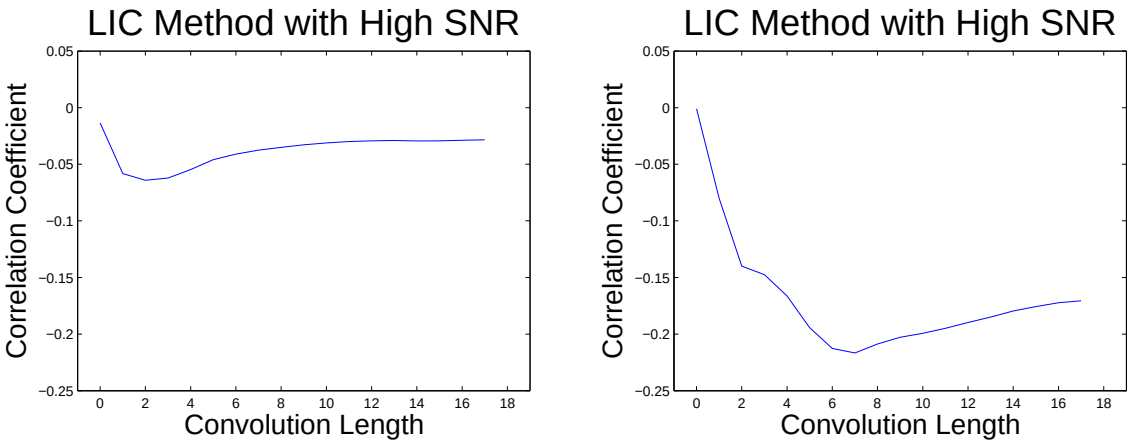

Figure 4.11. Change in correlation coefficient value for a range of convolution length on datasets with low and high SNR-values.

correlation values are computed again varying convolution length values. Note that there is an optimum value for each dataset with low and high SNR-values.

As shown in Figure 4.11, the correlation coefficient values change with the change in convolution length parameter. There is an optimum convolution length value for each dataset with different SNR-value. For small convolution lengths, effect on noise images is small. For large convolution lengths, noise image will be over modified. Therefore for both cases, correlation value between the output image and diffusion tensor data will be small. To analyze the best performance of LIC method, the maximum absolute value of correlation coefficients are examined for both dataset with high and low SNR-values. As is can be expected, high SNR-value dataset produces better correleation.

4.4. MR-DTI Application

DT-MRI is a medical imaging technique based on MR principles. The output of DT-MRI is a 2nd order symmetric positive semi-definite tensor (a 3*3 symmetric positive semi-definite matrix)that approximates the diffusion process at a given point. It was shown that the tensor is correlated with the underlying structure in restricted media such as nerve fibers in human brain. More specifically, it was shown that the principle eigenvector of this tensor (which is called principle diffusion direction, PDD) is parallel to the fibers. Thus, DT-MRI hac become popular for reconstructing fiber

structure (network) in human brain in vivo. The minor eigenvectors of this tensor are generally ignored which makes the technique rather questionable. Thus, visualizing all components of the diffusion tensor in 3D poses a significant problem which is still being investigated by researchers.

In this thesis, we tried to solve this problem by proposing two new methods. A sample MR-DTI data is used to test methods. Before the applying new methods, 3D volume of brain is visualized by using well-known methods in literature. For each tensor, ellipsoid that characterize the diffusion information is used by modifying the shape, orientation, color, and opacity properties of it. By the help of user interface, different slices are visualized. FA threshold value is used to display only tensors that has FA value is bigger than threshold. In figure 4.12, diffusion tensor field is colored by using anisotropy coefficients with two FA threshold value. Principle Diffusion Direction(PDD) is used for coloring in figure 4.13 for two FA threshold value.

As shown in Figure 4.14-17, four tensor painting methods are applied to MR-DTI data. The slice is visualized with basic visualization methods. Painted tensors are colored with white. Painting started from the regions where fiber population is high.

The results of LIC method are represented with output texture and local streamlines. In figure 4.18-19, the output texture images and streamlines that are generated for LIC method are visualized for different convolution length values.

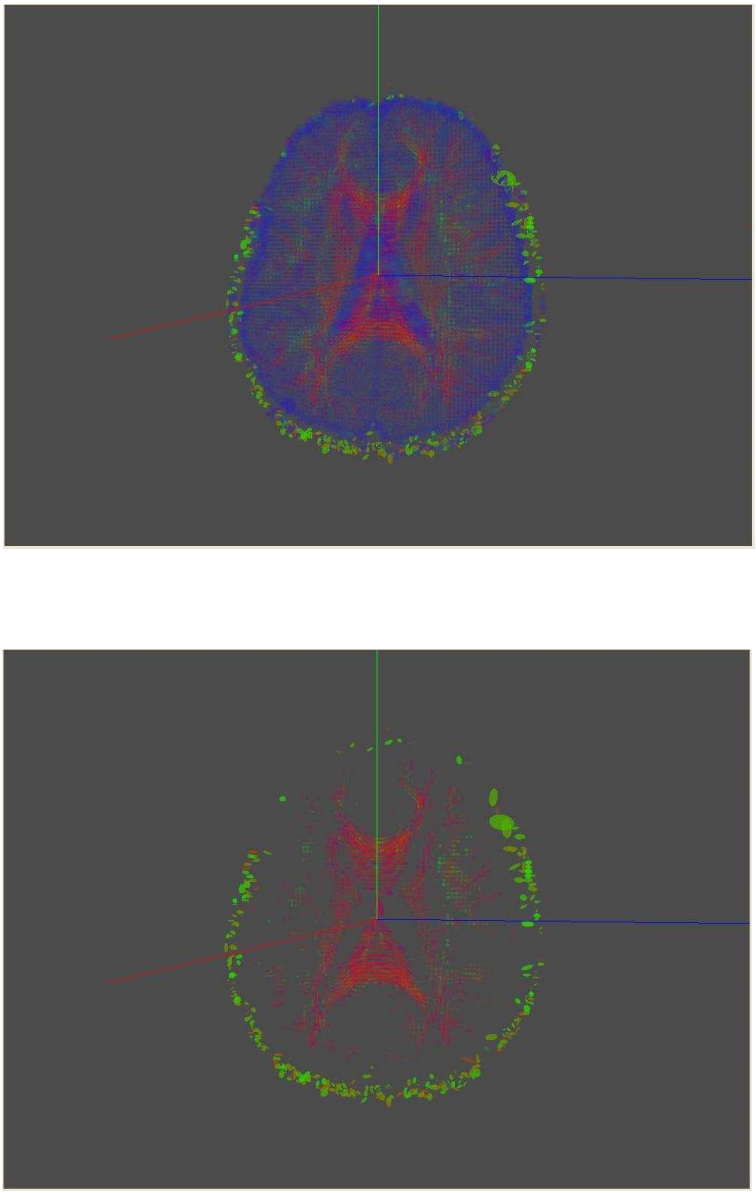

Figure 4.12. Basic Visualization Method with coloring based on anisotropy with Low(Top Image) and High(Bottom Image) FA threshold values

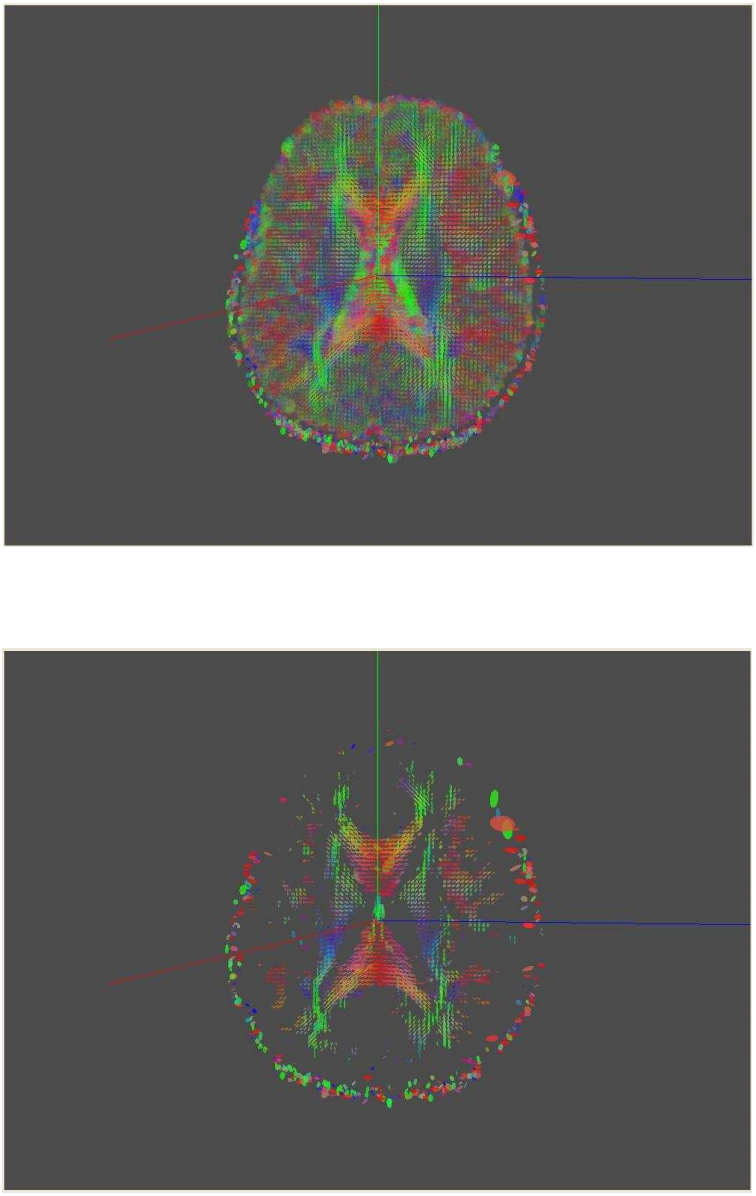

Figure 4.13. Basic Visualization Method with coloring based on PDD with Low(Top Image) and High(Bottom Image) FA threshold values

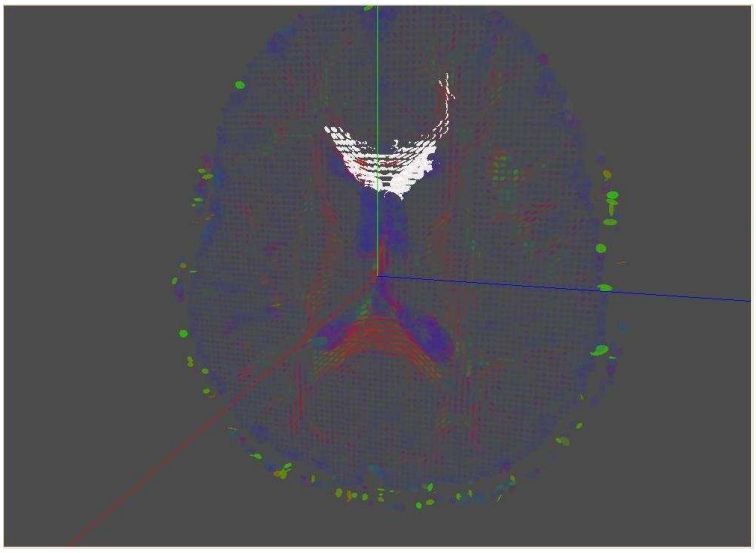

Figure 4.14. Tensor painting Results for Method 1

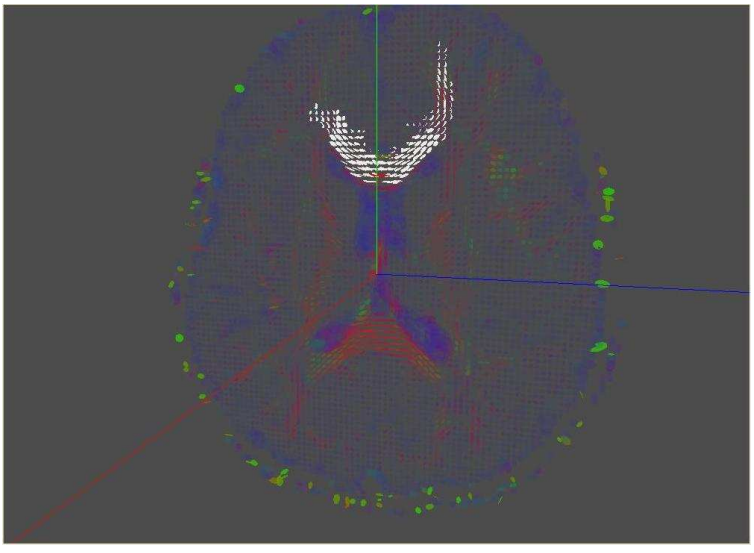

Figure 4.15. Tensor painting Results for Method 2

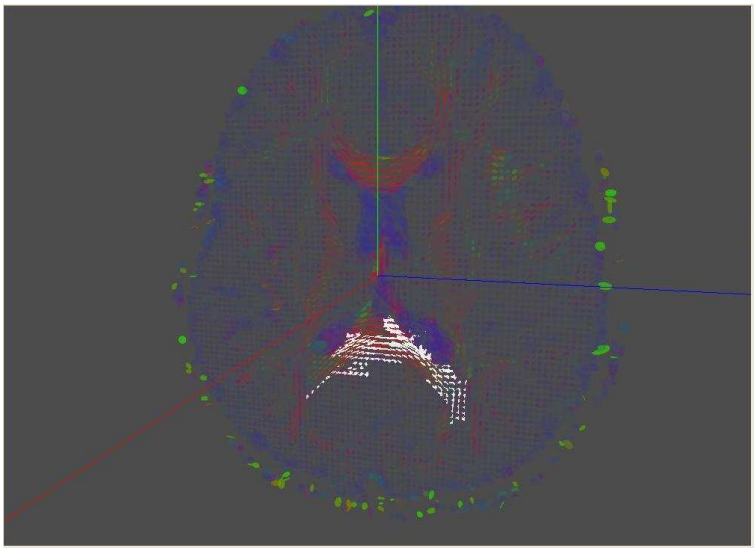

Figure 4.16. Tensor painting Results for Method 3

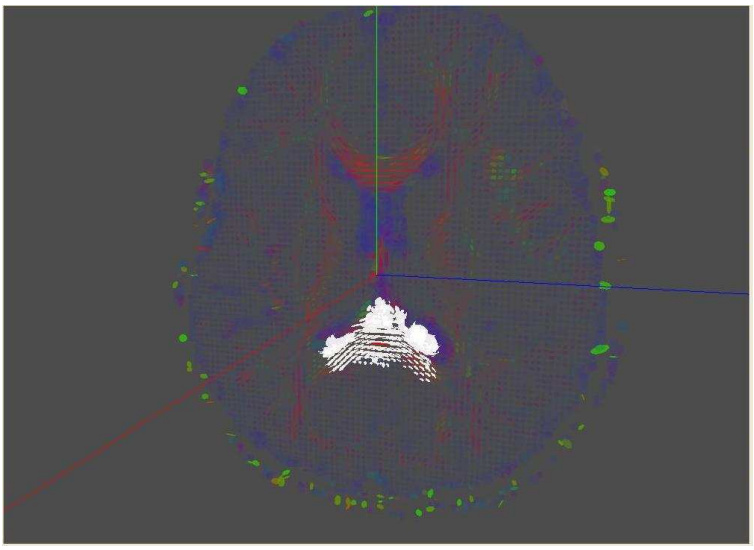

Figure 4.17. Tensor painting Results for Method 4

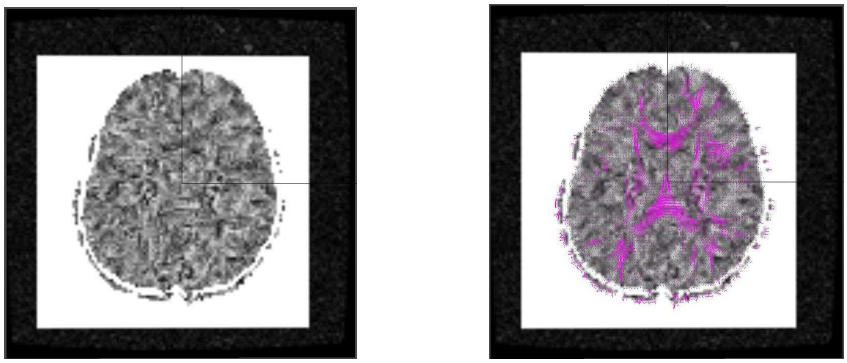

Figure 4.18. The LIC image with high Convolution Length and Stream Lines that are generated for LIC Method.

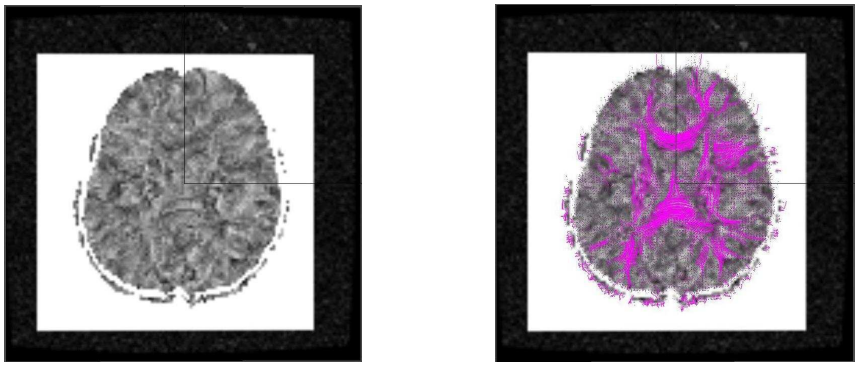

Figure 4.19. The LIC image with medium Convolution Length and Stream Lines that are generated for LIC Method.

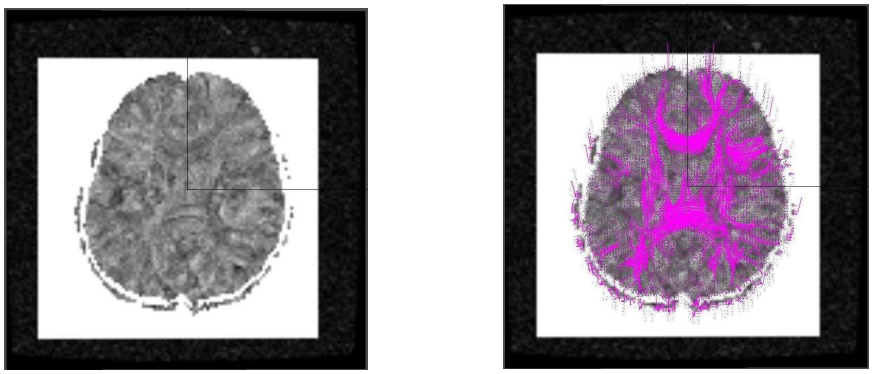

Figure 4.20. The LIC image with small Convolution Length and Stream Lines that are generated for LIC Method.

5. CONCLUSION

In this thesis, a tool has been developed for visualization of diffusion tensor fields. A number of basic visualization methods and two newly proposed methods are implemented in the tool. The user interface allows modification of different parameters for visualization purposes. The tool is implemented as a QT application, with OpenGL and C++. Algorithms and user interface are tested using phantom diffusion tensor field data. For different types of data and varying methods, evaluation criteria are calculated for a range of parameters.

Diffusion tensor field data is represented by six components for each point in 3D space. This multi-variate nature of data is the main difficulty in visualization. A common visualization goal is to depict the spatial patterns for the principle eigenvector only in regions where it is meaningful, rather than depicting all the tensor information. To visualize these patterns local connections and flow paths are extracted from the original data. And they are visualized via graphical objects. Another difficulty is to render these objects graphically with tolerable interaction rate. Adjustments are made in selecting the graphical objects and the interaction parameters. Evaluation of the methods posed another difficulty. Since there are no well-defined requirements for visualization, the evaluation of the methods are usually based on subjective criteria. We have tried to developed a set of objective evaluation methods based on mathematical properties of diffusion tensor fields.

We have also applied our methods to a set of real life DT-MRI data and tried to evaluate the visualizations empirically. It is our intention to develop the visualization tool further, with the purpose of employing it in direct medical usage. For this purpose, we are planning to design a DTI specific application that is based on VAVFrame[3]. This framework can cope with medical data in different formats and provides basic visualization functionalities. Diffusion tensor visualization tool will be integrated to this framework. Then, the visualization will be evaluated comparing the results to the requirements specified by specialists who use such data in medical applications.

REFERENCES

1. NIH/NSF, "Visualization Research Challenges, January 2006".

2. John H. Heinbockel, "Introduction to Tensor Calculus and Continuum Mechanics".

3. VAVLab Webpage ,"http://www.vavlab.ee.boun.edu.tr".

4. D.H. Laidlaw, E.T.Ahrens, D.Kremers, M.J. Avalos, C.Readhead, and R.E. Jacobs. "Visualizating diffusion tensor images of the mouse spinal cord", *in Proceeding of IEE Visualization 1998*, pages 127-134,1998.

5. C.-F. Westin, S. E. Maier, H. Mamata, A. Nabavi, F. A. Jolesz, and R. Kikinis. "Processing and visualization for diffusion tensor MRI". *Medical Image Analysis*, 6:93108, 2002

6. G. Kindlmann. "Superquadric Tensor Glyphs". *In Joint EUROGRAPHICS - IEEE TCVG Symposium on Visualization*, 2004.

7. G. Kindlmann, C-F Westin ,"Diffusion Tensor Visualization with Glyph Packing", *IEEE Transactions on Visualization and Computer Graphics* Volume 12, Number 5, Pages 1329-1335 September-October, 2006.

8. W. C. de Leeuw and J. J. van Wijk, "A probe for local flow field visualization",*In Visualization, 1993. Visualization 93, Proceedings., IEEE Conference on*, pages 3945, San Jose, California, USA, 1993.

9. W.Shen and A.Pang, "Anisotropy Based seeding for hyperstreamlines", *In IASTED Computer Graphics and Imaging (CGIM)*, Kauai, Hawaii, USA, 2004.

10. T. Delmarcelle and L. Hesselink. "Visualizing Second-Order Tensor Fields with Hyperstreamlines", *IEEE Computer Graphics and Applications*, 13(4):2533, 1993.

11. S. Zhang, Ç . Demiralp, and D. H. Laidlaw, "Visualizing Diffusion Tensor MR Images Using Streamtubes and Streamsurfaces", *IEEE Transactions on Visualization and Computer Graphics*, 9(4):454462, 2003.

12. S. Zhang, C. Curry, D. Morris, and D. Laidlaw, "Streamtubes and streamsurfaces for visualizing diffusion tensor mri volume images", *In IEEE Visualization 00 Proceedings, Work in Progress*, 2000.

13. L. Zhukov, A. H. Barr, "Oriented Tensor Reconstruction: Tracing Neural Pathways from Diffusion Tensor MRI", *In IEEE Visualization Proceedings of the conference on Visualization '02* , pages 387 - 394, 2002, Boston, Massachusetts.

14. G. Kindlmann, D.Weinstein, and D. Hart, "Strategies for Direct Volume Rendering of Diffusion Tensor Fields", *IEEE Transactions on Visualization and Computer Graphics*, 6(2):124138, 2000.

15. X. Zheng, A. Pang, "HyperLIC", *In IEEE Visualization 2003*, pages 249256, Seattle, Washington, USA, 2003.

16. B. Cabral, L.(Casey) Leedom, "Imaging Vector Fields Using Line Integral Convolution",*In SIGGRAPH '93: Proceedings of the 20th annual conference on Computer Graphics and interactive techniques*, pages 263-270, New York, USA, 1993, ACM Press.

17. S.C.L. Deoni ,B.K. Rutt, T.M.Peters, "Visualization of Neural DTI Vector Fields Using Line Integral Convolution", *Medical Image Computing and Computer-Assisted Intervention - MICCAI 2003*, 207-214, 2003.

18. A. Helgeland, O. Andreassen, "Visualization of Vector Fields Using Seed LIC and Volume Rendering",

19. I. Hotz, L. Feng , H. Hagen , B. Hamann, K. Joy,B. Jeremic, "Physically Based Methods for Tensor Field Visualization", *In IEEE Visualization Proceedings of the*

conference on Visualization '04, pages 123 - 130, 2004.

20. L. Hesselink, Y. Levy, and Y. Lavin, "The Topology of Symmetric, Second-Order 3D Tensor Fields", *IEEE Visualization Proceedings of the conference on Visualization '94*, pages 140 - 147, 1994, Washinton, D.C.

21. D. Weinstein, G. Kindlmann, and E. Lundberg, "Tensorlines: Advection-diffusion based propagation through diffusion tensor fields", *In IEEE Visualization 99 Proceedings*, pages 249 253, 1998.

22. G. Kindlman, D Weinstein, and D. A. Hart. "Strategies for direct volume rendering of diffusion tensor field", *IEEE Transaction on Visalization and Computer Graphics*, 6(2):124-138, 2000.

23. G.Kindlman, D Weinstein, "Hue-Balls and Lit-Tensors for Girect Volume Rendering of Diffusion Tensor Field",*In IEEE Visualization 99 Proceeding*, pages 183-189, 1999.

24. B. Acar, "DTI-MRI Connectivity And/Or Tractography? Two New Algorithms",*Similar NoE Tensor Workshop*, Las Palmas, November 2006.

25. B. Acar, R. Bammer, M. E. Moseley, "Comparative Assessment of DT-MRI Fiber Tractography Algorithms", *Proceedings of ISMRM 2003*, Jul. 2003, Toronto, Canada.

26. Song Zhang, David H. Laidlaw, Gordon Kindlmann ,"Diffusion Tensor MRI Visualization", Chapter 16, *Visualization Handbook*, 2005.

27. A. Sigfridsson, T. Ebbers, E. Heiberg, Lars Wigstrom, "Tensor Field Visualization Using Adaptive Filtering of Noise Fields Combined with Glyph Rendering".

28. Yoruk E., Acar B., Bammer R., "A Lattice of Springs Framework for DT-MRI Based Connectivity Mapping".

29. Yoruk E., Acar B., Bammer R. , "A Physical Model for DT-MRI Based Connectivity Map Computation", *Lecture Notes in Computer Science*, Vol. 3749 / Part 1, pp. 213-220, Springer, 2005 (MICCAI 2005, Palm Springs, CA.

30. Qt Webpage, "http://trolltech.com/products/qt".

31. D. Hearn and M. Pauline Baker, "Computer Graphics with OpenGL (3rd Edition)".

32. N. Dale, D. Teague,"C++ Plus Data Structures (2nd edition)".

33. T. Lidy,"FLOWVIZ","http://www.cg.tuwien.ac.at/courses/Visualisierung/2001-2002/Ergebnisse/Beispiel2/LidyT/index.html"

34. E.R. Tufte, "The Visual Display of Quantitative Information", Graphics Press, September 2002, Cheshire, Connecticut.