Exploring brain connectivity with two-dimensional maps

Çağatay Demiralp, Radu Jianu, and David H. Laidlaw

Abstract We present and compare two low-dimensional representations, *2D point* and *2D path*, for tractography datasets. The goal is to facilitate the exploration of dense tractograms by reducing visual complexity both in static representation and during interaction. The proposed planar maps have desirable properties such as visual clarity, ease of tract-of-interest selection, and multiscale hierarchy. The 2D path representation also conveys the anatomical familiarity of 3D brain models and cross-sectional views. We demonstrate the utility of the representations in two interactive systems where the views and interactions of the standard 3D streamtube representation are linked to those of the planar representations. We also demonstrate a web interface that integrates precomputed neural-path representations into a geographical digital-maps framework with associated labels, metrics, statistics, and linkouts. We compare the two representations both anecdotally and quantitatively via expert input. Results indicate that the planar path representation is more intuitive and easier to use and learn. Similarly, users are faster and more accurate in selecting bundles using the path representation than the 2D point representation. Finally, expert feedback on the web interface suggests that it can be useful for collaboration as well as quick exploration of data.

1 Introduction

Diffusion-weighted MRI (DWI) enables neural pathways in the *in vivo* brain to be estimated as a collection of space curves, called a tractogram. The study of tractograms (i.e., tractography) has important applications in both clinical and basic neuroscience research on the brain. Tractograms have visual complexity proportional to the intricacy of the axonal brain connectivity and, with increasing DWI resolutions, this complexity is becoming greater and greater. It is thus often difficult

Çağatay Demiralp · Radu Jianu · David H. Laidlaw
Brown University; e-mail: {cad,jr,dhl}@cs.brown.edu

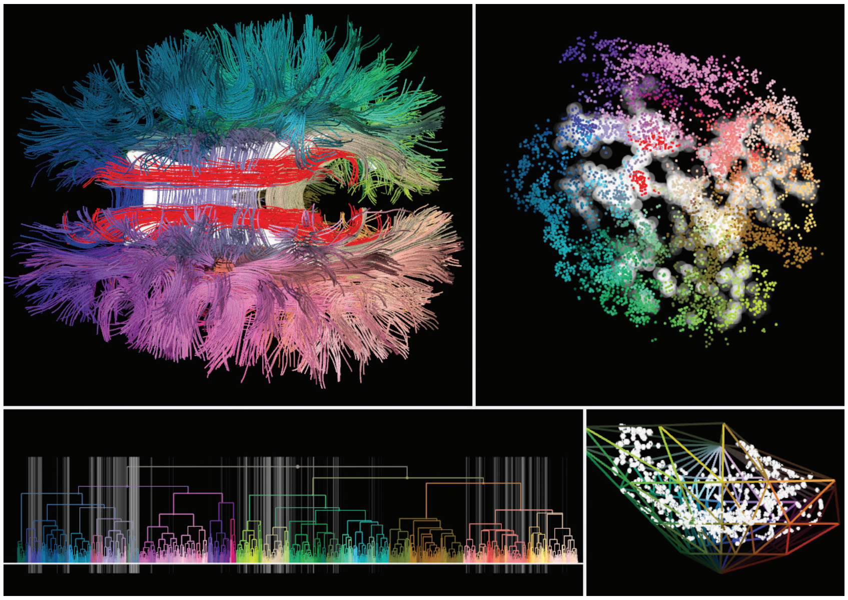


Fig. 1: *2D point* representation linked with a streamtube representation in an interactive tractography visualization tool.

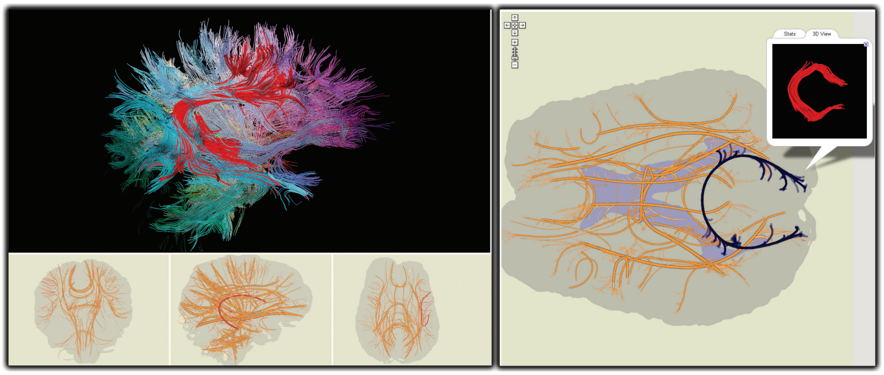


Fig. 2: Schematic *2D path* projections of tractograms as part of a standalone interactive system (left) and as a web-accessible digital map (right). The digital map interface easily incorporates any tract-associated information, including labels, links, metrics, and statistics. Shown in the pop-up window on the right is the "brain view" of the selected tract.

for practitioners to see tract projections clearly or identify anatomical and functional structures easily in these dense curve collections. This is important because, for example, a clinical study into a neurodegenerative disease typically involves selecting more than 30 tract of interests (TOIs) manually across different datasets. Therefore, it is necessary for tractography visualization tools to provide means to reduce and help cope with visual complexity at interaction and representation levels. We believe two concepts, *abstraction* and *filtration*, applied to representation of datasets can help users overcome the difficulties of visual complexity. While abstraction involves simplification and generalization, filtration here entails clustering and hierarchization. With these ideas in mind, we proposed low-dimensional point representations for better interaction with fiber tracts in [17], along with [7]. Driven by known embedding methods, embedding in two-dimensional space provides an interesting window into the manifold space of neural connectivity and help in fine selection of tracts. A drawback of point representations is, however, that coordinate axes in the low-dimensional space lack an anatomical interpretation. It is clear from evaluations in [7, 17] that having a frame of reference, anatomical or otherwise, is important for users. Motivated by this problem, we introduced two-dimensional neural paths that have the desirable properties of low-dimensional representations while preserving meaningful and familiar coordinates in [18]. In this chapter, we provide a unified discussion of our these earlier work on planar point and path representations of tractograms [17, 18].

DWI is the source of all the data used in the work presented here. We first give an elementary introduction to DWI and then briefly discuss related work on tractography and web-based visualizations.

2 Related Work

Diffusion-weighted magnetic resonance imaging (DWI) measures the diffusion rate of water molecules in biological tissues *in vivo* [24]. Since tissue characteristics, geometric or otherwise, at a given point affect the diffusion rate, measured diffusion-rate information is an indicator of the tissue characteristics at the point. In particular, water in fibrous tissues such as brain white matter (i.e., a collection of myelinated axons) diffuses faster along fibers than orthogonal to them. Therefore, it is possible to estimate fiber trajectories computationally using diffusion models such as the tensor model that quantify anisotropic diffusion. Diffusion imaging based on fitting second-order tensors to DWI sequences is known as diffusion-tensor magnetic resonance imaging (DTI) [4]. Fiber trajectories are computed from DTI data by integrating bidirectionally along the principal eigenvector of the underlying tensor field. This process, called fiber tracking, yields a dense collection of integral curves (i.e., a tractogram). All the tractograms used our work were obtained by fiber-tracking in DTI volumes.

Tractograms are often visualized with streamlines or variations of streamlines in 3D [22, 28]. Reflecting the intricacy of the connectivity in the brain, these 3D models are generally visually dense. Consequently, typical interaction tasks over tracts, such as fine bundle selection, are often difficult to perform and have been a focus of recent research [1, 2]. We proposed planar point representations to improve interaction with DTI fiber tracts [17], concurrently with Chen et al. [7]. There are, however, differences between these two works. First, we use hierarchical clustering to create multiscale representations, which makes the exploration of large datasets easier, both visually and computationally. Second, while our work uses a simple force-based embedding method, Chen *et al.* uses the SMACOF algorithm, an iterative method that minimizes the metric stress of multidimensional scaling (MDS) using majorization [21]. And third, we use the embedding procedure also to create a "nice" coloring such that colors of data points perceptually reflect the structure in data points.

To address the expert concern about the insufficient anatomical context in the 2D point representation, we recently introduced 2D path representations [18], which are also projections of fiber tracts into a plane, but as planar curves rather than points.

One of the advantages of the 2D path representation is that it can be naturally integrated into web-based digital geographic map frameworks. Basic data visualization has been available on the web for many years but was usually limited to traditional techniques such as bar graphs and charts. More recently, however, visualization research started targeting this environment and advanced applications have emerged. ManyEyes [26] paved the way for everyday data visualization, with subsequent studies such as [25] and [8] proving the need for accessible web visualization. While web-development toolkits such as [5] greatly aid web visualization development, large-scale web-visualization is limited by inherent browser capabilities, as demonstrated in [19]. Alternatively, stand-alone systems have been made available in applet form or can be run as client applications directly from websites. However, users still must control the parameters involved in producing visualizations, specify their data queries and learn the system features. This often constitutes an undesirable overhead. Yet another approach, most similar to our work from an implementation standpoint, is to use Ajax (asynchronous JavaScript and XML) technology to perform the rendering on the server side and serve images asynchronously to the client browser. The essential difference between the present work and traditional offline visualization systems is that the former separates interaction and display from rendering and computation. Our brain maps differ by eliminating users' efforts in creating visualizations and assigning this task to experts and by using the Google Maps API, which is an Ajax framework for interactive display of pre-rendered images. Closest to our work are X:MAP [27] and Genome Projector [3], which present genome browser tools implemented using the Google Maps API. We extend this idea to a new domain and demonstrate its usefulness for tractography datasets.

3 Methods

Both point and path representations are projections of fiber tracts onto the plane: Each tract is represented with a 2D point in the former and with a 2D curve in the latter. Formally, given a polyline form of a fiber tract $C \in \mathbb{R}^{n\times 3}$ with n vertices in 3-space, its point representation is obtained with a map $\pi_{point} : \mathbb{R}^{n\times 3} \to \mathbb{R}^2$ and its path representation is obtained with another map $\pi_{path} : \mathbb{R}^{n\times 3} \to \mathbb{R}^{m\times 2}$.

Generation of these two representations shares three common steps. First, we obtain a whole-brain tractogram by fiber tracking in a diffusion-tensor volume fitted to a given DWI brain sequence. Second, we compute similarities between all pairs of tracts within the tractogram, obtaining a similarity (or affinity) matrix. Third, using the similarity matrix from the previous step, we run a hierarchical clustering algorithm on the tractogram, obtaining a clustering tree (i.e., dendrogram).

We create the 2D point representation of the tractogram by embedding the tracts in the plane with respect to the similarity matrix, using a simple iterative force-directed method. We use the hierarchical clustering tree to create multiscale point representations.

In the case of the path representation, we first pick a cut on the clustering tree and obtain a clustering. Then, by treating cluster centroids as pivots, we create projections of tractograms onto the major orthogonal planes as curves. We render these 2D curves stylistically using heuristics determined by the topology and geometry of the corresponding tracts and tract clusters.

We give details of these steps in subsequent sections.

3.1 Image Acquisition and Fiber Tract Generation

DWI brain datasets used in this paper were acquired from healthy volunteers on a 1.5T Siemens Symphony scanner with the following acquisition parameters in 12 bipolar diffusion-encoding gradient directions: thickness = 1.7 mm, FOV = 21.7 cm × 21.7 cm, TR = 7200 ms, TE = 156 ms, b = 1000, and NEX = 3. For each DWI sequence, the corresponding DTI volume was then obtained by fitting six independent parameters of a single second-order tensor at each voxel to the twelve measurements from the DWI sequence [4]. We generate fiber-tract models of the whole brain by integrating (second-order Runge-Kutta integration) the major eigenvector field of the diffusion tensor field bidirectionally starting at seed points. We integrate with a constant step size of 0.5 mm and stop integration when a gray-matter area is reached or the linear anisotropy or signal-to-noise (SNR) ratio at the current point passes predetermined thresholds.

3.2 Measuring Similarities Between Fiber Tracts

We quantify the similarity between two tracts using the distance measure discussed in [9]. This measure tries to capture how much any given two tracts follow a similar path, while giving more weight to the points closer to tract ends. Given two integral curves $C_i = \{C_i^1, \ldots, C_i^m\}$ and $C_j = \{C_j^1, \ldots, C_j^n\}$ that are represented as polylines with m and n vertices respectively, we first find mean weighted distances d_{ij} and d_{ji}, and then determine the maximum of these two distances as the distance D_{ij} between the two curves:

$$d_{ij} = \frac{1}{m} \sum_{k=1}^{m} {\alpha_i}^k \mathrm{dist}(C_i^k, C_j) \tag{1}$$

$$d_{ji} = \frac{1}{n} \sum_{k=1}^{n} {\alpha_j}^k \mathrm{dist}(C_j^k, C_i) \tag{2}$$

$$D_{ij} = D_{ji} = \max(d_{ij}, d_{ji}) \tag{3}$$

The function $\mathrm{dist}(p, C)$ returns the shortest Euclidean distance between the point p and curve C. Also, $\alpha_k = \frac{1}{Z} e^{|k-(m+1)/2)|^2/\sigma^2}$, where the normalizing factor $Z = \sum_{k=1}^{m} e^{|k-(m+1)/2|^2/\sigma^2}$. We set the parameter σ automatically, proportional to L_C the length of the fiber tract, such that $\sigma = \lambda L_C$, where $\lambda \in (0,1]$. We set $\lambda = 0.5$ for the datasets used for this paper.

We compute distance between each pair of integral curves as we denoted and assemble the measures to create a distance matrix. Note that our measure is symmetric and positive-definite but does not necessarily satisfy the triangle inequality, therefore, it is not a metric. While our approach is independent of a particular similarity measure, good results in practice require a good similarity measure – one that reflects users' understanding of the similarity between data points (i.e., tracts) and works well for the task at hand.

3.3 Clustering

For a given tractography dataset we compute a clustering tree using an average-linkage hierarchical clustering algorithm on the tract distance matrix (e.g., [10]). We choose the average-linkage criterion because it is less sensitive than the minimum-linkage to broken tracts due to tracking errors. The output of the clustering algorithm is a hierarchical tree called dendrogram. The height of the tree can be thought as the radius of the bounding ball of the dataset–in the units of the similarity measure used. And any horizontal cut on this tree provides a clustering of the dataset. Therefore, for example, the root node represents a clustering with a single cluster containing all the data points. Conversely, the leaf nodes correspond to a clustering where every data point is a cluster.

We obtain a clustering of tracts by manually setting a cut threshold on the dendrogram. This threshold can be also interactively changed by user to control the coarseness of the clustering. A constant cut at 60% of the clustering tree's height gave consistent results across the six datasets we experimented with.

3.4 Planar Projections of Fiber Tracts

3.4.1 Fiber Tracts as Embedded Points

We use a simple iterative force-directed method for embedding tracts in the plane [11]. Embedding refers to a one-to-one smooth mapping from fiber tracts to points in the plane that preserves the "structure" of the fiber tracts. In this context, the distance matrix computed can be considered as a manifestation of the structure in the fiber tract space.

For a given dataset with M tracts, we start with M corresponding points, all placed at the origin initially. We then iteratively adjust the positions of these points by moving the pairs of points closer to or further from each other to match with the corresponding D_{ij} entries in the similarity matrix. To achieve an interactive performance, we use a stochastic sampling technique described in [6] for updating the "forces" between data points. Briefly, instead of computing forces on a point x_i from every other point in the dataset, we limit the points acting on x_i to $x_j \in \mathscr{F} = \{N_i \cup S_i\}$, where N_i and S_i are disjoint sets with a constant size. We iterate over data points and resample S_i each time by uniformly randomly selecting points from the whole dataset. For a randomly selected point x_k, if D_{ik} is smaller than $\max_{x_l \in N_i} D_{il}$, the maximum distance from x_i to any member of N_i, then x_k is assigned to N_i, otherwise it is assigned to S_i. N_i loosely represents the neighborhood of the point iterated.

3.4.2 How Embedding Can Be Used to Color Fiber Tracts

Given a similarity measure, a good coloring of fiber tracts should reflect the similarities between the tracts such that similar tracts are assigned to similar colors and different tracts are assigned to different colors. Embedding fiber tracts in perceptually-uniform color spaces, which are subsets of $\mathbb{R}^3$, provides a practical way to approximately achieve this goal. A perceptually uniform color space is an empirically-constructed color space where the Euclidean distances between color triplets are approximately proportional to perceptual differences between them. L*a*b* and Luv are two common examples of such color spaces [12].

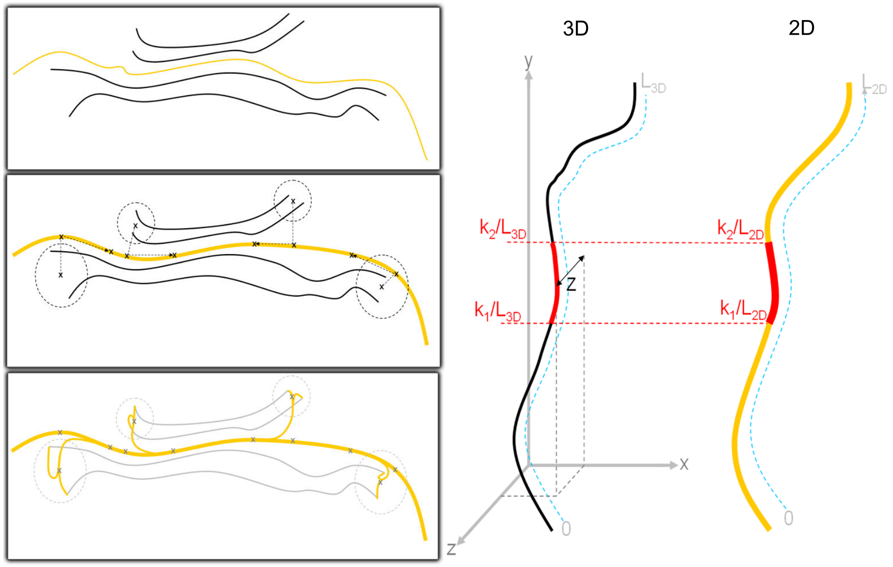
Embedding fiber tracts in the L*a*b* space is the general coloring scheme used in the interactive tools presented here. For this, we compute an approximation of the L*a*b* color gamut, as visible on the right panels of Figure 6, and use it as a container for force directed embedding. To avoid having to adjust a repulsive container force, which would likely need a hard-to-control, steep gradient, we perform a physically accurate simulation with container contact detection. The embedding begins in the center of the gamut and is gradually expanded until most of the space is filled. During implementation we observed that the largest distances are often embedded along the luminance axis, the vertical (in paper coordinate frame) axis, of the color gamut. This is problematic because luminance offers little resolution and can be interpreted as a lighting effect. We therefore apply a “flattening” force at the beginning of a simulation cycle to force large distances to lie in the horizontal plane (a*b*-plane). These force components, acting on the luminance axis towards the center of the gamut, wear off as the embedding moves towards a steady state. The force computation used is the same as for the 2D embedding, with straightforward 3D modifications. In terms of interaction, the color embedder only supports collapsing and color grabbing.

3.4.3 Fiber Tracts as Planar Curves

For a given tractography dataset, we create schematic views of fiber tracts projected on the sagittal, coronal, and transverse planes.

We start with obtaining a clustering of the dataset by picking a cut on the hierarchical clustering tree computed earlier. We then create simple orthogonal projections of tracts on each plane. Suppose the sagittal plane is aligned with the xy-plane. And let $v = (x, y, z) \in \mathbb{R}^{n\times 3}$ be a vertex of a tract. Then the projection onto the sagittal is given by this simple equation $\pi(v) = (x, y)$. We cull out tracts that do not contribute significantly to the projection. If the ratio of projected tract length to true tract length is under a threshold value, we remove the tract from the corresponding cluster. We set the culling threshold to 0.65 for the projections used in our experiments. Finally, we compute a centroid for each cluster by choosing the tract with the smallest maximum distance to any other tract in the cluster. We found that for illustration purposes it is desirable to avoid broken tracts. We therefore weigh the centroid selection to favor longer tracts by dividing the maximum distance from each tract to any other tract by the tract’s length.

We opted for a non-photorealistic rendering of brain projections to avoid their interpretation as 3D views and to harness user’s intuition about 2D maps. The rendering assumes a given clustering with assigned centroid tracts, which can be computed as described in the previous section. The centroid tracts will define a schematic neural skeleton on top of which the non-centroid tracts are scaffolded. Projections of centroid curves are smoothed prior to rendering to achieve a schematic representation and to reduce clutter. This is done by sampling a number of evenly distributed control points (5 in our implementation) along the tract projection and using them as control points for a a piecewise cubic spline with 30 segments. The thickness of a centroid curve is proportional to the square root of the number of tracts in the bundle. Once centroid tracts are represented as 2D splines, endpoints of non-centroid curves are linked to their cluster’s centroid spline following the procedure illustrated in Figure 3a. First, the end-points of non-centroid curves in a bundle are clustered based on the end-points’ initial 3D coordinates. Two endpoints are placed in the



(a) (b)

Fig. 3: (a) Schematic tract-cluster representation. (Top) 2D projections of a tract-bundle, with an associated centroid curve (orange), are determined from a hierarchical clustering of initial 3D tracts. (Middle) The centroid curve is smoothed by a spline and the endpoints of non-centroid curves are clustered using their initial 3D coordinates (four clusters); for each cluster, three control points linking the center of the cluster to the centroid spline are computed. (Bottom) Splines are run from each curve endpoint through the control points of its corresponding cluster. (b) Depth ordering of 2D paths. For each segment of a 2D spline, we locate a corresponding segment on the 3D curve from which the spline was derived by traveling the same fractional distance along both curves. The depth of the 2D segment is the same as the depth of the middle of its corresponding 3D segment.

same cluster if the distance between them is less than 2 mm. Then, for each such endpoint cluster we compute three control points that link the geometrical center of the endpoint cluster to the centroid spline: the first point is the center itself, the second is a point on the centroid spline closest to the center point, and the third is determined by traveling from the second point down the centroid spline, towards each curve's other endpoint, for a predefined distance (e.g., half of the distance between the first two points). Ultimately, splines are run from each tract endpoint through its cluster's three control points, thus linking each endpoint to the centroid path. The thickness of these endpoint linkage splines gradually increases from unit thickness (i.e., single-tract thickness) at the tract endpoint to a thickness proportional to the square root of the endpoint cluster size, where it merges with the centroid spline.

We depth-order spline segments so that 2D centroid splines crossings can indicate the depth ordering of their corresponding 3D shapes. The depth ordering is done differently for centroid splines and non-centroid splines, since while centroid curves

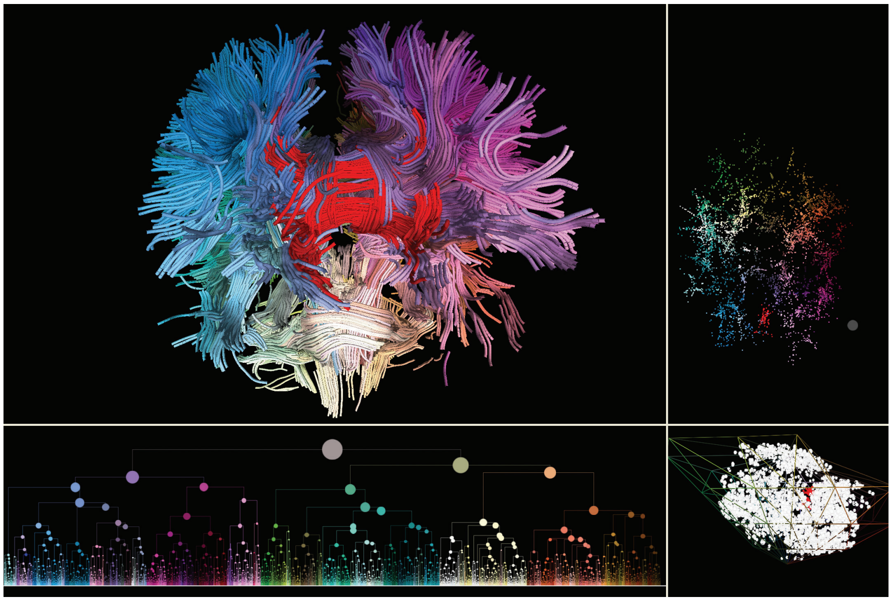


Fig. 4: Coordinated DTI tractogram model exploration in lower dimensional visualizations: 2D embedding (upper-right), hierarchical clustering (lower-left), and L*a*b* color embedder (lower-right). A selection of a fiber-bundle (red) in the hierarchical clustering is mirrored in the other views.

are close representations of actual 3D tracts, non-centroid curves are abstract representations obtained through the process described above. Furthermore, the depth ordering is approximate (as discussed in the following paragraph) and may produce artifacts. For centroid splines, the depth of a spline segment is computed by finding a matching segment on the 3D tract from which the spline was derived, and taking the depth of that segment's center (see Figure 3b). The matching segment on the 3D tract has its endpoints at the same fractional distance from the start of the 3D tract as the 2D segment's distance from the start of the 2D spline. The depth of any non-centroid spline is determined by averaging the depth of the corresponding 3D tract.

In the following two sections, we give details on how we use 2D neural path representations as part of an interactive application and as standalone digital maps.

3.5 *Linked Multi-view Interaction*

We expect a typical use of low-dimensional representations to be as part of interactive applications where views and interactions of conventional representations are linked with that of low-dimensional representations. We have developed two interac-

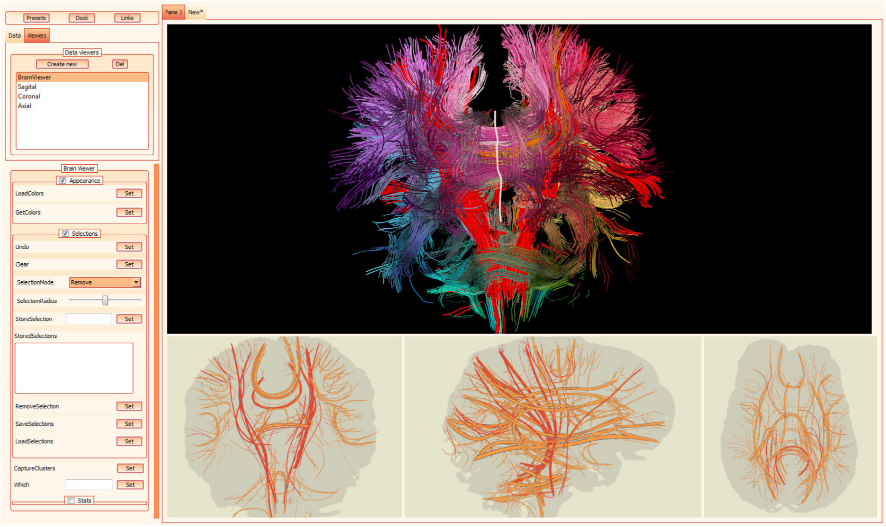


Fig. 5: An interactive analysis system using linked views and planar tract-bundle projections. Three planar representations, along the coronal, transverse and sagittal planes (bottom panels), are linked to a 3D streamtube model (upper left) and a 2D point embedding of tract similarities (upper right). Selections in the projection views can be performed by clicking or cutting across cluster curves and are mirrored in the 3D view. Points corresponding to the selected tracts are interactively embedded into the plane and used to refine selections at tract level.

tive visualization systems using the 2D point and path representations, respectively, to demonstrate this mode of use (see Figures 3.5 and 3.5).

Both applications have a view of tractography data visualized using 3D streamtubes. Coloring generated through the embedding of tract similarity into the L*a*b* color space. In addition to the standard 3D viewing interactions, we have two basic 3D selection/deselection interactions on streamtube models: sphere-selection and brushing. Sphere-selection, like box-selection, enables user to select the intersecting tracts by moving a sphere of desired radius. Brushing lets user draw 2D curve on the viewing plane and select the intersecting tracts.

Both sphere-selection and brushing can be used to further prune the current selection but they cannot be used to grow it. For that purpose, we provide a selection growing interaction that gradually adds tracts closest to the current selection. Proximity is determined again by the distance measure discussed above.

On the 2D point representation, we provide point selection, point collapsing. Selection is performed by clicking and dragging; multiple selection can be performed to select points from non-adjacent regions. Collapsing groups a set of points into a single clustered representation. This can be used either for easier tract bundle selection or as a mechanism for manually refining embeddings–points belonging to the same tract bundle can be grouped together if the embedding algorithm places them apart. The centroid of the grouping will be used in subsequent embedding iterations.

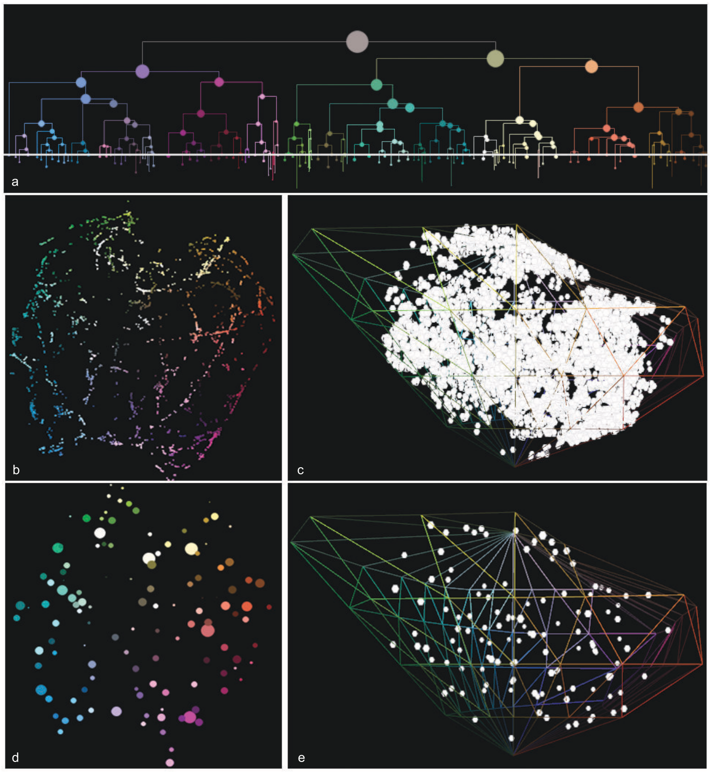


Fig. 6: A clustering cut in the dendrogram view (top row) is applied to the linked 2D embedding and 3D colorer (middle row). Points belonging to the same cluster are collapsed to their centroids (bottom row).

The hierarchical clustering tree provides a filtration of the dataset via cuts. Figure 6 shows how a cut on the dendrogram, which results in a clustering, provides a coarser representation. Both the 2D embedding and the 3D colorer clusterings can be further refined interactively after a cluster cut, by expanding or collapsing individual clusters.

In the 2D path tool, we link projective views on the sagittal, coronal, and transverse planes to a standard 3D streamtube model. The clustering cut threshold that defines the specificity of the projected bundles can be altered interactively during visualization. Tract clusters in the planar projections can be selected by drawing line segments that select intersecting bundles. A selection in any of the planar views is mirrored in the 3D model view as well as all other 2D projections.

3.6 Digital Map Interface

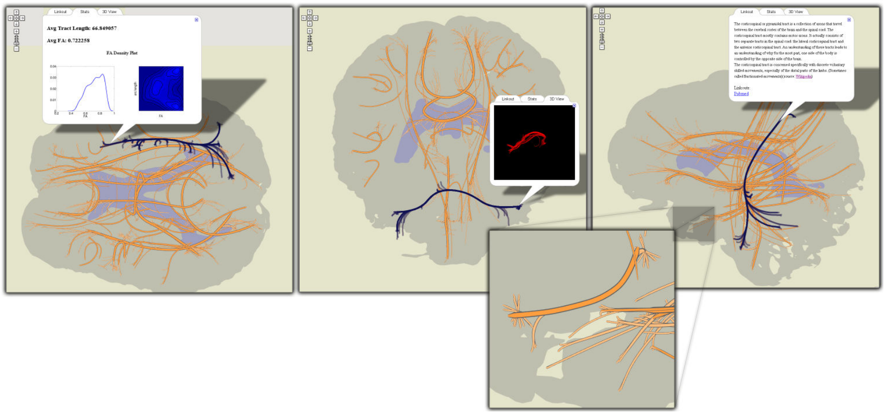
Brain mapping is one of the quintessential problems in neurosciences. We believe that a geographical map metaphor is well-suited to the visualization and analysis of results obtained in that direction. Therefore, having a representation of the brain that is viewed, interacted, queried, and enriched like an online geographical map was one of the motivations behind our creation of 2D path representation.

For this, we use the Google Maps API, an Ajax framework used to render large maps, to interactively display our tractogram maps on the web. The Google Maps API receives input image data in the form of a set of small images, called tiles, that when assembled together form the different zoom levels of the map. Each zoom level, z, consists of a rectangular grid of tiles of size $2^z \times 2^z$. The API decodes the zoom level and coordinates of the currently viewed map region to retrieve and display the visible tiles. The developer can load a custom set of tiles in the API by implementing a callback function that translates numerical tile coordinates and zoom level into unique paths to the custom tiles.

The API provides basic functionality such as zooming and panning and allows programmatic extension or customization with markers and polyline overlays, information pop-ups and event management. The API can easily be integrated into any webpage supporting Javascripts.

Our visualization system can render our 2D projections into a set of image tiles instead of the screen. For each cluster, including both tract-bundled and endpoint clusters, we export information required for interaction and browsing. Selection information consisting of evenly spaced points along splines and thickness radii for splines contained in a cluster is exported. In line with the tile paradigm, instead of exporting this information to a single large file, we divide it geometrically across corresponding tiles and write it as multiple tile-content text files. Upon user selection, the content file of a clicked tile is fetched from the server and its data analyzed for an intersection. This approach avoids loading and searching through large files. A valid cluster selection is marked on the map with polyline overlays running over tract splines contained in the selected cluster (see Figure 7). For this purpose, spline coordinates for each cluster are exported to files indexed by a unique cluster identifier.

Finally, for each tract cluster we export a variety of metadata accessible during map browsing in information boxes, as shown in Figure 7. A short description and links to the most relevant publications or research can be manually added for major tracts. A few 3D poses of each tract bundle are prerendered and exported as animated GIF images, indexed by the cluster identifier. Statistical data, in both textual and graphical form, are computed for each cluster and written as HTML content to cluster indexed files. This information is loaded and displayed in tabbed information boxes at the user's request.



Avg Tract Length: 66.849057

Avg FA: 0.722258

The corticospinal or pyramidal tract is a collection of axons that travel between the cerebral cortex of the brain and the spinal cord. The corticospinal tract mostly contains motor axons. It actually consists of two separate tracts in the spinal cord: the lateral corticospinal tract and the anterior corticospinal tract. An understanding of these tracts leads to an understanding of why for the most part, one side of the body is controlled by the opposite side of the brain.
The corticospinal tract is concerned specifically with discrete voluntary skilled movements, especially of the distal parts of the limbs. (Sometimes called fractionated movements)(source: Wikipedia)

Linkouts:
Pubmed

Fig. 7: DTI tractography data projected onto the sagittal, coronal and transverse planes. Major tract bundles are represented schematically by their centroid tract; individual tracts in bundles are linked from the centroid bundle to their projected end points. Zooming in allows access to smaller clusters of tracts. Bundles can be selected and pre-computed statistical data along with 3D views of the tract bundle ("brain view") can be displayed.

3.7 Implementation

We implemented both interactive systems in C++ using G3D and Qt libraries [13, 23]. We created the web interface for the neural path representation using the Google Maps API [14]. It can be accessed via the url link [29].

4 User Evaluation

We compared the two representations both anecdotally and quantitatively.

4.1 Anecdotal Study: Methods and Results

In the anecdotal study we showed a prototype that implements both 2D point and neural path representations to 3 neuropsychologists. They were all interested in the relationship between fiber tracts and cognitive and behavioral function in the brain. Similarly, all have used computational tools for analyzing DTI data, though only one of them has used fiber-tract visualization tools in his clinical research. The participants had research interests in vascular cognitive impairment, early Alzheimer's disease, and HIV, focusing on specific tracts and regions such as the corpus callosum

(CC), frontal lobe, basal ganglia, cingulate bundle, superior and inferior longitudinal fasciculi, anterior internal capsule, and the uncinate fasciculus.

Our anecdotal evaluation protocol was straightforward: we demonstrated the prototype while asking questions and collecting participant's feedback. Two of the experts also tried both interfaces themselves by selecting a set of major TOIs, the CC, cingulate bundle, uncinate anterior internal capsule, and the corticospinal tract. There was agreement that the 2D neural path representation was more intuitive and easier to use and learn than the 2D point representation.

Our experts also found the web interface with the digital map interaction useful. Although they believed that the standalone application with linked representations would remain necessary for quantitative analyses that require interactive fine selection, they thought the web accessibility opened up interesting possibilities. They were particularly excited about browsing through datasets while commuting or at home, of quickly inspecting unfamiliar datasets, and of sharing such visualizations with collaborators.

4.2 Quantitative Study

In the quantitative evaluation, we compared the point and path representations by measuring user performance on a bundle selection task.

4.2.1 Applications

We used our interactive path representation system (Section 3.5) and the 2D point representation tool presented in [7] for running the comparative study. The reason that we used this tool instead of ours is that it was already compared to other known tractography tools such as CINCH, MedINRIA, and BrainApp, and reported to be preferable over them. Using it would give us some idea about, if any, relative merits of the 2D path representation tool over not only the 2D point representation but also these other tractography applications compared before. Chen *et al.*'s application offers a brush tool that works similarly to ours in 3D and as a lasso tool on the 2D point representation. Users can select tracts or points and then remove them or, conversely, remove everything else from a current selection.

4.2.2 Participant Pool

We had 4 subjects all familiar with neuroanatomy and tractography. They also had experience with one or more tractography visualization tools. Our first subject was a neuroscience graduate student, working on tracing white-matter tracts from frontal subregions to basal ganglia and the medial temporal lobe. Our second user was a postdoc in neuropsychology and had five years of experience with DWI in clinical research. This user, who participated in the anecdotal study as well, studied white matter variation with neurodegenerative diseases as specified above. Our third subject was a biomedical engineering graduate student and had significant tract-selection experience working as a rater for a neuroscientist. Our last subject was a computer-science graduate student doing research on computational DWI algorithms. Two of the users were male and two female.

4.2.3 Task

We measured user performance on bundle selection, a typical real world task in tractography tools. Users were asked to select three major bundles, the cingulate bundle (cb), corticospinal tract (cst), and right superior longitudinal fasciculus (slf), in two different brain datasets. We choose these bundles because they represent the easy-to-hard selection-difficulty range well and were used for evaluation in [7].

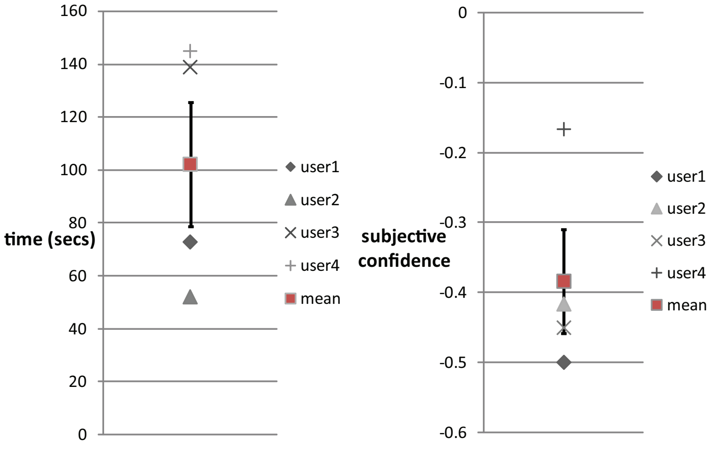
For each system, we explained to users the underlying visualization concepts and demonstrated the basic interactions, mainly involving brushing on 2D and streamtube representations. After this introduction, users were asked to select the bundles (cb, cst, and slf) on two different training datasets. Following training, the users performed the task on two different test datasets while we collected their task-completion times. After each selection they provided subjective confidence estimate in the range 1-5 (1: not confident, 5: very confident) for their selection. They could give fractional estimates. After completing the task on both systems, users were asked to complete a post-questionnaire giving qualitative feedback on their experience. Half the users performed the task first on the 2D point-representation tool and the other half on the 2D path tool.

4.2.4 Factors and Measures

The sole factor considered in our quantitative experiment was the type of low-dimensional representation: 2D point and 2D path. All subjects used both types of representation. We recorded the users' bundle-selection times and subjective confidence values as measures of performance.

4.2.5 Results

In order to understand if the differences between user performances on the two tools were significant, we ran the paired t-test on our measurements. Results show that users were significantly faster on the 2D path tool than the 2D point tool ($p = 0.02$). Users were also significantly more confident with using the 2D path representation than the 2D point representation ($p = 0.01$). Table 1 summarizes users' overall and per-bundle mean performances on each tool. Figure 4.2.5 shows the difference between the means of performance measures per user (2D-path-performance values



(a) (b)

Fig. 8: Per-user differences between (a) time and (b) confidence measurements with the two tools. Differences are obtained by subtracting 2D-point-tool performance values from 2D-path-tool performance values. Red squares show the mean performance difference between the tools. Errors bar around the red squares indicate the standard error of per-user differences.

	time (secs)				confidence			
	cb	cs	slf	mean	cb	cst	slf	mean
2D point	227	361	234	274	4.1	3.3	3.1	3.5
2D path	136	165	215	172	4.1	3.8	3.7	3.9

Table 1: User performance on bundle selection task.

are subtracted from 2D-point-performance values) and the mean over users. Errors bars indicate the standard error of per-user differences.

We observed some interaction patterns worth reporting. We noticed two distinct selection strategies used with the 2D path tool. Two of the users consistently brushed over large areas of the projection to ensure that the targeted bundle was selected and then relied on the 3D view to clean up the selection. The other two users aimed for fine selections in the 2D projections and then inspected the 3D view to determined whether any fibers the selection. They added the missing tracts using short, targeted brush strokes and then removed tubes that were erroneously added during this operation. These users seemed to have a better understanding of the mapping between the 3D view and the 2D projections, perhaps explaining the difference in strategies.

All subjects used the 2D point representation relatively rarely. The most common operation was to remove points they were completely confident were not part of the selection (e.g., half of the brain, or peripheral U-shaped bundles). However, in the absence of a clear contextual mapping between the 2D point and streamtube views, subjects were hesitant to perform bold operations in 2D, at least in the short run.

5 Discussion

It is important to note that our representations rely on the anatomical fidelity of the intermediate results at each step. For example, broken trajectories due to fiber tracking errors can reduce the effectiveness of the representation. Similarly, our method expects the clustering algorithm and similarity measure to provide anatomically plausible results. However, it is difficult for a single distance measure to capture the anatomical similarity completely. Furthermore, on the same data, a good similarity measure for one purpose can be entirely irrelevant for another. While the choice of similarity measure makes clustering a subjective task, clustering algorithms themselves also have intrinsic limitations [20].

One potential limitation of the planar path representation is that bundles surrounded by the other bundles similar in orientation and shape may not be clearly visible. While we have not found this to be an issue in practice, moving projection planes along major axes while restricting the projecting tracts to a volumetric window moving with the projection plane can help solve potential problems. Also, results of user studies should be taken with a grain of salt. In general, it is difficult to run experiments that vary one factor while keeping all the other factors constant. For example, an earlier study [7] compared the 2D point representation tool with other tractography applications and reported that users were faster with the former. However, in our user evaluation, we observed that users rarely used the 2D point representation and the brush tool dominated their interaction. This brings up an open question of whether the performance difference in the reported evaluation was due mainly to the brush tool or to the 2D point representation. An experiment that replaced the brush tool with a more standard box-selection tool, say, might resolve this question. In either case, we believe that abstract representations, including the 2D point representation, are useful in the long run, as users gain more experience with the mapping between brain tractograms and low-dimensional representation primitives. In general, however, it is not realistic to expect practitioners to learn the correspondence between the new representation and the actual fiber-tract collection quickly, unless the tools are easily interpretable using a conventional anatomical framework. Furthermore, in order any tool using a new representation of tractograms to have a clinical relevance, it should provide anatomical context and intuitive functionality for region of interest (ROI) analysis on both conventional and new representations.

While we have focused on planar spatial representations here, it is possible to create abstract representations of tractograms. For example, the hierarchical clustering tree itself can be considered as a representation of the tractogram. Or consider the circular map of connectivities (or dependencies) shown in Figure 9. We obtain this dependency graph representation by first clustering tract end points using hi-

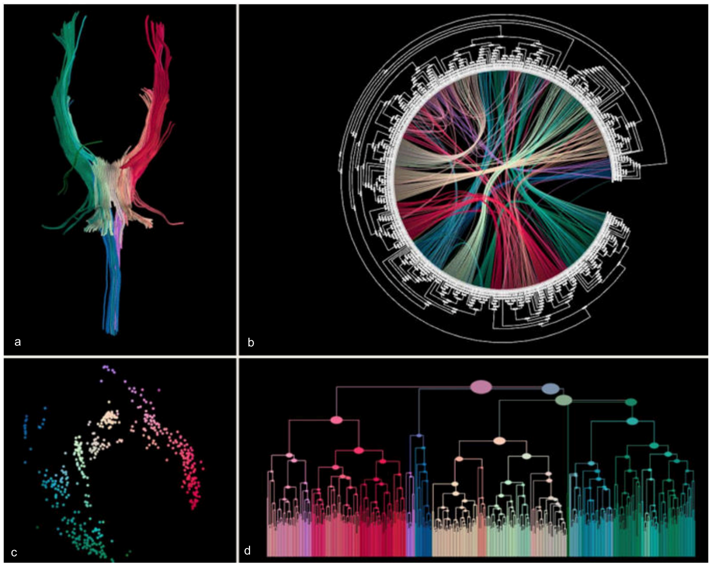


Fig. 9: Low-dimensional representations can have varying degree of abstractions. The internal capsule represented with (a) streamtubes, (b) a circular connectivity map, (c) a set of embedded points in the plane, and (d) a hierarchical clustering tree.

erarchical clustering and setting an implicit dependency between the end points of each tract. We then visualize the resulting hierarchical tree with pair-wise connectivities using hierarchical edge bundling [15]. Feedback from a neuropsychologist suggest that it might be useful for understanding connectivity densities and profiles.

Although DWI is the only imaging protocol to estimate the brain neural architecture *in vivo*, there are *in vitro* imaging techniques, such as the three-dimensional electron microscopy particularly used in the emerging field of connectomics, with which it is possible to extract neural structures on much smaller scales (e.g., individual axon bodies) [16]. We believe that the general ideas as well as specific techniques presented in this paper can extend to the visualization and analysis of visually complex axonal structures originating from these high-throughput imaging techniques.

6 Conclusions

Combining traditional 3D model viewing with intuitive low-dimensional representations with anatomical context can ease navigation through the complex fiber tract models, improving exploration of the connectivity in the brain. We have presented two planar maps, point and path representations of tractograms, that facilitate exploration and analysis of brain connectivity. Both representations are essentially applications of abstraction and filtration concepts to tractograms. We achieve abstraction by simplifying and generalizing, both geometrically and topologically, fiber tracts with points and schematic curves in the plane. We create filtrations of tractograms by computing hierarchical clustering trees. These help create better abstractions as well as provide a multiscale view of data, which is important in reducing visual complexity and noise. We compare the two representations both qualitatively and quantitatively with help of experts. Results suggest that the 2D path representation is more intuitive and easier to use and learn than 2D point representation.

We also introduced a novel way of making tractography data accessible by publishing neural maps online through a digital map framework. Our representation is conducive to such a geographic map interface by construction. This interface leads to new possibilities for enriching tractography datasets using the mass knowledge base available on the web. User feedback indicates that our web interface can be particularly useful for browsing unfamiliar datasets quickly, for analysis tasks that do not require fine selection and for sharing data in collaborative settings.

Acknowledgments

We thank Song Zhang for generously providing us the 2D point representation tool used for comparison in our quantitative evaluation. This work was supported by NIH grant 1R01EB00415501A1.

References

1. Akers, D.: Wizard of Oz for participatory design: Inventing an interface for 3D selection of neural pathway estimates. In: Proceedings of CHI 2006 Extended Abstracts, pp. 454–459 (2006)
2. Akers, D., Sherbondy, A., Mackenzie, R., Dougherty, R., Wandell, B.: Exploration of the brain's white matter pathways with dynamic queries. In: Proc. of Visualization, pp. 377–384 (2004)
3. Arakawa, K., Tamaki, S., Kono, N., Kido, N., Ikegami, K., Ogawa, R., Tomita, M.: Genome Projector: zoomable genome map with multiple views. BMC bioinformatics **10**(1), 31 (2009)
4. Basser, P.J., Mattiello, J., LeBihan, D.: Estimation of the effective self-diffusion tensor from the NMR spin echo. J Magn Reson B **103**(3), 247–254 (1994)
5. Bostock, M., Heer, J.: Protovis: A graphical toolkit for visualization. IEEE Transactions on Visualization and Computer Graphics **15**(6), 1121–1128 (2009)
6. Chalmers, M.: A linear iteration time layout algorithm for visualising high-dimensional data. In: Proceedings of the 7th conference on Visualization'96. IEEE Computer Society Press Los Alamitos, CA, USA (1996)
7. Chen, W., Ding, Z., Zhang, S., MacKay-Brandt, A., Correia, S., Qu, H., Crow, J.A., Tate, D.F., Yan, Z., Peng, Q.: A novel interface for interactive exploration of DTI fibers. IEEE TVCG (Proc. of Visualization) (2009)
8. Danis, C., Viegas, F., Wattenberg, M.: Your place or mine? Visualization as a community component. In: Proceedings of CHI (2008)

9. Demiralp, C., Laidlaw, D.H.: Similarity coloring of DTI fiber tracts. In: Proceedings of DMFC Workshop at MICCAI (2009)
10. Duda, R.O., Hart, P.E., Stork, D.G.: Pattern Classification, 2nd edn. Wiley-Interscience (2000)
11. Eades, P.: A heuristic for graph drawing. Congressus Numerantium **42**(149160), 194–202 (1984)
12. Fairchild, M.D.: Color Appearance Models. WileyBlackwell (2004)
13. G3D. http://g3d-cpp.sourceforge.net/
14. GoogleMapsAPI. http://code.google.com/apis/maps/
15. Holten, D.: Hierarchical edge bundles: Visualization of adjacency relations in hierarchical data. IEEE Transactions on Visualization and Computer Graphics **12**, 741–748 (2006). DOI http://doi.ieeecomputersociety.org/10.1109/TVCG.2006.147
16. Jeong, W.K., Beyer, J., Hadwiger, M., Vazquez, A., Pfister, H., Whitaker, R.T.: Scalable and interactive segmentation and visualization of neural processes in EM datasets. IEEE Transactions on Visualization and Computer Graphics **15**, 1505–1514 (2009)
17. Jianu, R., Demiralp, C., Laidlaw, D.: Exploring 3D DTI fiber tracts with linked 2D representations. IEEE TVCG (Proc. of Visualization) **15**(6), 1449–1456 (2009)
18. Jianu, R., Demiralp, C., Laidlaw, D.H.: Exploring brain connectivity with two-dimensional neural maps. In: IEEE Visualization 2010 Poster Compendium (2010)
19. Johnson, D., Jankun-Kelly, T.: A scalability study of web-native information visualization. In: Proceedings of Graphics Interface, pp. 163–168 (2008)
20. Kleinberg, J.M.: An impossibility theorem for clustering. In: NIPS, pp. 446–453 (2002)
21. de Leeuw, J.: Applications of convex analysis to multidimensional scaling. In: J. Barra, F. Brodeau, G. Romier, B.V. Cutsem (eds.) Recent Developments in Statistics, pp. 133–146. North Holland Publishing Company (1977)
22. Mori, S., Van Zijl, P.: Fiber tracking: principles and strategies–a technical review. NMR in Biomedicine **15**(7-8), 468–480 (2002)
23. Qt. http://www.qtsoftware.com/
24. Tanner, J.E.: Transient diffusion in system partitioned by permeable barriers. Application to NMR measurements with a pulsed field gradient. Journal of Chemical Physics **69**(4), 1748–1754 (1978)
25. Viégas, F., Wattenberg, M., McKeon, M., Van Ham, F., Kriss, J.: Harry Potter and the meat-filled freezer: A case study of spontaneous usage of visualization tools. In: Proc. HICSS (2008)
26. Viegas, F., Wattenberg, M., Van Ham, F., Kriss, J., McKeon, M.: ManyEyes: a site for visualization at internet scale. IEEE Transactions on Visualization and Computer Graphics **13**(6), 1121 (2007)
27. Yates, T., Okoniewski, M., Miller, C.: X: Map: annotation and visualization of genome structure for Affymetrix exon array analysis. Nucleic Acids Research **36**(Database issue), D780 (2008)
28. Zhang, S., Demiralp, C., Laidlaw, D.: Visualizing diffusion tensor MR images using streamtubes and streamsurfaces. IEEE TVCG **9**(4), 454–462 (2003)
29. http://graphics.cs.brown.edu/research/sciviz/newbraininteraction/