

DTI in Context: Illustrating Brain Fiber Tracts In Situ

Pjotr Svetachov, Maarten H. Everts, and Tobias Isenberg

University of Groningen, The Netherlands

Abstract

We present an interactive illustrative visualization method inspired by traditional pen-and-ink illustration styles. Specifically, we explore how to provide context around DTI fiber tracts in the form of surfaces of the brain, the skull, or other objects such as tumors. These contextual surfaces are derived from either segmentation data or generated using interactive iso-surface extraction and are rendered with a flexible, slice-based hatching technique, controlled with ambient occlusion. This technique allows us to produce a consistent and frame-coherent appearance with precise control over the lines. In addition, we provide context through cutting planes onto which we render gray matter with stippling. Together, our methods not only facilitate the interactive exploration and illustration of brain fibers within their anatomical context but also allow us to produce high-quality images for print reproduction. We provide evidence for the success of our approach with an informal evaluation with domain experts.

Categories and Subject Descriptors (according to ACM CCS): I.3.3 [Computer Graphics]: Picture/Image Generation—Line and curve generation

1. Introduction

Illustrative visualization is a new research direction that bridges research from traditional illustration to computer-supported visualization [VGH*05, ES06]. The medical domain is an important application area due to its centuries of experience with illustrations. To this day, hand-drawn and, in particular, pen-and-ink illustrations are being used not only in medical textbooks but also in scientific papers due to their potential to clearly show important aspects of the depicted objects, e. g., through abstraction and emphasis.

We are inspired by this continued use of pen-and-ink techniques in medical illustration (e. g., [Dau05]). We demonstrate how pen-and-ink rendering of the brain can be used to show internal brain structures such as fiber bundles extracted from DTI data [MvZ02] within their anatomical context [WJNP*04]. Medical researchers often combine computer-generated visualizations of fiber structures (to show detail) with images that show where these fibers are located within the brain's anatomy (e. g., [CTdS08] and conversations with a neuroscientist). Sometimes these two types of visualizations are separate and/or use a hand-drawn context (e. g., [SPW*07]). It is from these practical observations with domain experts that we derive our motivation for the work presented here: by combining detail and context in one single image using pen-and-ink rendering, we provide a means for

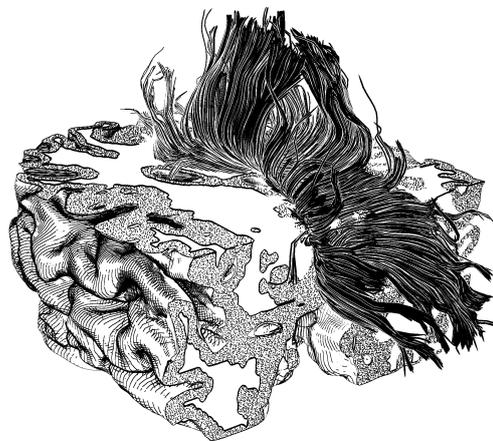


Figure 1: Example of visualizing DTI fiber tracts in situ.

these researchers to explore and illustrate brain fiber tracts in context in a clear and understandable way and at interactive frame rates (e. g., Fig. 1).

For this purpose we use surface models of the context structures, obtained either from interactive iso-surface extraction based on MRI data [TSD07] or from segmentation data, and render these using a slice-based hatching tech-

nique [DHR*99]. This hatching is guided by screen-space ambient occlusion [SA07]. The resulting hatching style can be used on triangular surfaces without connectivity information and without the need to extract, for example, curvature information. In addition, we ensure that the amount of hatching depends on the chosen resolution to make it scale-dependent for a consistent appearance over a range of zoom levels [SALS96, FMS01]. We also use cutting surfaces and construct intersections onto which we render gray matter with stippling [KCODL06]. These techniques are combined with an illustrative visualization of DTI brain fiber tracts [EBRI09] in screen space. These fiber tracts are visually separated from the context/background using an additional halo around them. Finally, we explore different appearances for context elements [TIP05] to further enhance the depiction. Our contribution, therefore, consists of describing how to combine several previously developed non-photorealistic and illustrative rendering techniques such that they can be used provide structural context for the pen-and-ink visualization of DTI fiber tract data. Using our GPU implementation, the resulting illustrative visualizations can be rendered and controlled at interactive frame-rates and also permit the generation of high-quality images for print reproduction.

The remainder of this paper is structured as follows. Section 2 details work related to our own. Section 3 then describes our approach for pen-and-ink rendering of brain fiber tracts within the context of the brain and skull. Next, we discuss some aspects of visual emphasis of focus & context and show a number of application scenarios in Section 4. Finally, Section 5 reports on an informal evaluation of the presented techniques before Section 6 concludes the paper and mentions some possible avenues for future work.

2. Related Work

This work takes inspiration from a number of disciplines including traditional pen-and-ink illustration, non-photorealistic rendering (NPR), illustrative visualization, and, more generally, scientific visualization. We outline the connections to these fields in more detail below.

2.1. Traditional Pen-and-Ink Illustration

Traditionally, pen-and-ink illustration[†] has been playing an important role in illustration in general and medical illustration in particular (e. g., [HP60, Dau05]). One important reason for this is the easy reproduction of pen-and-ink images in print because they use only one ‘color’—black [Hod03]. Illustrators use ink marks in form of dots, short strokes, or longer strokes to create outlines, form, and shading of the depicted objects. Combined with the use of illustration principles such as abstraction and emphasis, pen-and-ink is a pow-

[†] We consider techniques such as woodcuts and copperplates as part of pen-and-ink because they are based on similar principles.

erful medium for medical illustration (e. g., Fig. 2). We used this technique as the basic inspiration for our work (this is discussed in more detail in Section 3.1).

2.2. Non-Photorealistic Rendering

Techniques for pen-and-ink rendering have been created as one of the fundamental methods of NPR. The solutions range from early 2D approaches [WS94, Ost99] to 3D techniques [DHR*99, HZ00, ZISS04]. Also, techniques have been explored that focus on shorter marks [SFWS03] or that produce results in real-time using a texture-based approach [PHWF01]. While we are similarly interested in fast rendering methods, we also require dedicated control of the hatching lines. In particular, we need a scale-dependent hatching technique [SALS96, FMS01] to be able to maintain the same apparent intensity of a region for different zoom-levels.

Another important style element in NPR is the use of halos [ARS79, Elb95] which can greatly enhance depth perception [TCM06, BG07, EBRI09]. We use halos, in particular, to separate focus objects from the background. A technique roughly related to halos is the use of ambient occlusion [ZIK98, Lan02]. This technique is used to give a better impression of the shape of complex objects and has been applied to, for example, the visualization of molecules [TCM06]. We employ ambient occlusion to determine which parts of the outer brain’s surface are rendered using hatching.

2.3. Illustrative Scientific Visualization

A number of possibilities to use non-photorealistic rendering techniques for scientific and, in particular, medical visualization have been explored in the past [ER00, NSW02]. In particular, it was shown how to extract silhouettes or feature lines from volume data [YC04, BKR*05], how to generate hatching patterns to better visualize surfaces or vessel structures [TC00, DCLK03, RHD*06], and how to use stippling for volume visualization [LMT*03]. While we also use hatching and stippling as techniques, we focus on using these methods to show context for illustrative brain fiber tract visualizations [EBRI09], being inspired by the characteristics of hand-drawn anatomical illustrations.

This goal of providing context for objects in focus is an important aspect of our approach and has previously been explored in a number of visualization techniques. For example, it was demonstrated [TIP05] that different rendering techniques (shading, lines, direct volume rendering) can be combined to show focus, near-focus, and context for medical visualizations. Jaine et al. [JBB*08] presented a system that allows to illustratively visualize and explore brain activation data together with the brain’s anatomy and fiber tracts [BJH*09] while Nimsky et al. [NGE*06] combine traditional tube-based fiber tract rendering [ZDL03] with traditional volume rendering. Also related to our own work, Schultz et al. describe virtual Klingler dissection [SSA*08]

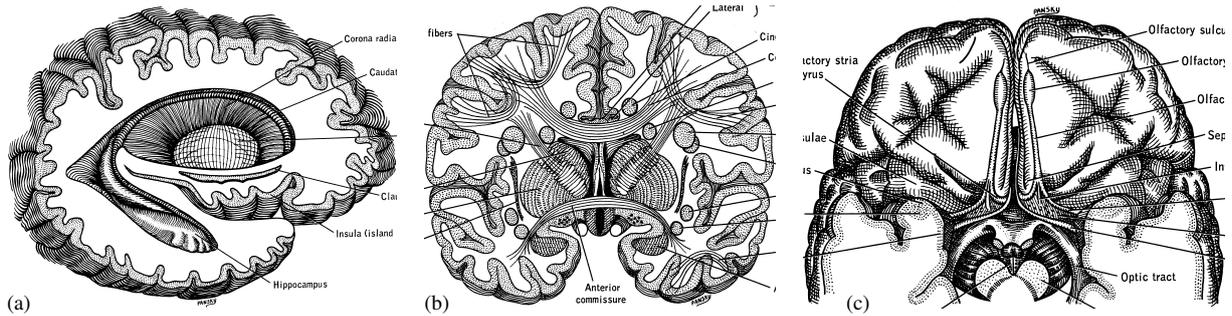


Figure 2: Hand-drawn pen-and-ink illustrations of the brain, from [HP60], © 1960 McGraw-Hill, used with permission.

that shows brain fibers within the context of the brain, albeit using deformed cutting planes and shaded fibers.

3. Pen-and-Ink Rendering of Brain Fibers and Context

Our overall goal is to visualize fiber structures together with context information using pen-and-ink style. For this purpose we use fiber tracts extracted from DTI datasets, matching MRI scans, and segmentation data created using specialized software. Based on this collection of data we perform three major steps: hatching the surface that we extract for the skull, the brain, and/or tumors, computing cutting surfaces and adding stippling for gray matter, and combining these context visualizations with an illustrative fiber rendering technique. These steps are described in detail below, but first we identify the visual aspects that we try to replicate from traditional medical pen-and-ink illustration.

3.1. Motivation and Identification of Goals

While we do not aim at exactly replicating hand-drawn pen-and-ink illustrations, we do take inspiration and a number of goals from those examples, including the ones shown in Fig. 2. In these illustrations we can observe, e. g., the use of hatching to show the shape of the brain’s surface while cutting surfaces are mostly white. An interesting observation is the use of hatching lines parallel to the cutting surface (as opposed to guided by the local curvature, Fig. 2(a), (b)), while for line-based rendering usually the use of principal direction is suggested (e. g., [IFP97, GIHL00]). Fig. 2(c) shows a more direct view on the brain’s outer surface. Here we can also see that the principal direction does not seem to be the only guidance; the fact that lines are generated roughly parallel to each other per hatching layer seems to be more important. In addition, we can observe that mostly regions that are close to silhouettes or regions around sulci are rendered using hatching lines, while other regions of the brain’s surface remain white. These observations together seem to suggest that ambient occlusion may be a good measure to guide the hatching, and that a stable vector field for placing roughly parallel lines is essential.

In Fig. 2(b) the illustrator augmented the cutting plane il-

lustration with a set of fibers. We can observe that the space between fibers often obstructs elements behind them, in particular, in the middle of the fiber bundles. While this example only uses a few lines, the lines representing fibers in the center of Fig. 2(a) are quite dense and sometimes even merge. We aim at using this type of fiber depiction and combining it with the brain illustration as in Fig. 2(b).

Finally, we observe that in these examples gray matter is shown using stippling, sometimes with a contour separating it from white matter (Fig. 2(a) and (b)) and sometimes without (Fig. 2(c)). The function of the stippling is to represent the type of brain matter using a very even tone, as opposed to also representing different shades, which is used when stippling is employed for surface rendering.

3.2. Hatching Surfaces of Medical Data

As a first part of our approach we describe the rendering of the general context, the surface of the brain, the skull, and/or tumors, using hatching. For this purpose we extract iso-surfaces or use segmentation data, employ a modified slice-based hatching technique, and use ambient occlusion to parametrize the rendering of the generated hatching lines.

3.2.1. Surface Extraction

Based on medical volumetric data, we use thresholding to specify an iso-surface that we want to visualize. Specifically, we employ skull-stripped MRI data because in MRI data without additional segmentation information the skull cannot easily be singled out using thresholding. Then, we extract the specified iso-surface using a real-time GPU technique [TSD07]. Because this operation is still relatively expensive we cache the output of the geometry shader using OpenGL transform feedback and re-use it while the threshold is constant. We only re-compute the iso-surface when the user changes the threshold value. As an alternative to this iso-surface extraction, it is also possible to use pre-segmented volume data, such as for the skull or for tumors.

3.2.2. Slice-Based Hatching

For rendering the extracted surfaces using roughly parallel hatching lines, we employ a method similar to the one pre-

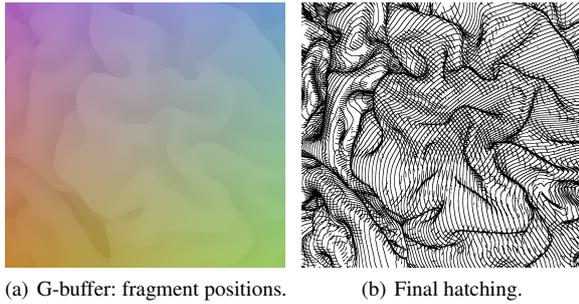


Figure 3: Slice-based hatching using the GPU.

sented by Deussen et al. [DHR*99]. Their method is based on cutting a polygonal model with skeleton-aligned cutting planes and extracting the intersection lines. We modify this general approach to make it suitable for GPU rendering as well as to allow the use of more flexible cutting surfaces. The approach works, in general, by determining the distance d_p of a given fragment F_i at location P_i to the defining cutting plane (located at $(0,0,0)$ with a normal vector N) as $d_p = P_i \cdot N$. The distance d_l of the fragment to its closest ('lower') cutting plane is then computed as d_p modulo a line separation distance d . Now we compute for this fragment F_i the distance of this 'lower' cutting plane to the defining cutting plane as $d_c = d_p - d_l$. Then, we check in F_i 's 8-neighborhood whether another fragment is 'above' or 'below' d_c . If this is the case, a black pixel is generated.

In practice we use deferred shading [DWS*88, HH04]. We first render the whole scene to a G-buffer [ST90] that stores the position of each fragment (Fig. 3(a)). This is re-used for several layers of cross-hatching. The positions are stored as 32 bit floats because a lower precision produces jagged lines when zooming in. Then we render this buffer to the screen and let a fragment shader calculate the fragments that are cut using the cutting planes as explained above (Fig. 3(b)). While this requires a re-computation of the fragments' distance to the cutting planes due to the 8-neighborhood lookup, it allows us to compute any number of cross-hatching layers in one pass. This pixel-based line generation ensures that all hatching lines have the same width—if only a fragment's distance to the closest intersection were used the hatching lines would get wider as the cutting planes become increasingly parallel to the iso-surface [Lei94]. We can also extend the size of the neighborhood look-up beyond the direct neighbors to permit line width control. This still only requires the lookup of 8 pixels in the G-buffer because the potential artifacts are typically negligible.

Hatching is a technique that is scale-dependent and for which the detail needs to be adapted according to the chosen zoom-level, assuming a constant width of the hatching lines. We follow the lead of Salisbury et al. [SALS96] and Freudenberg et al. [FMS01] by adding more hatching lines when one zooms in or removing lines when one zooms out. Ideally, one would gradually blend in a new distinct layer of

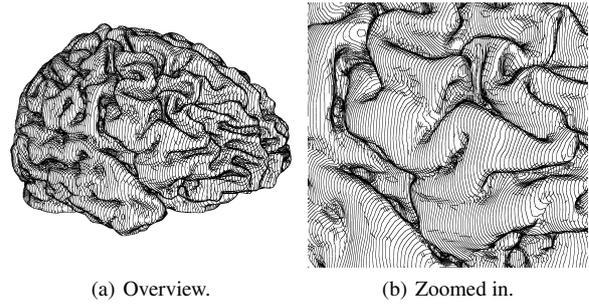


Figure 4: Adaptation of the line density to the zoom level.

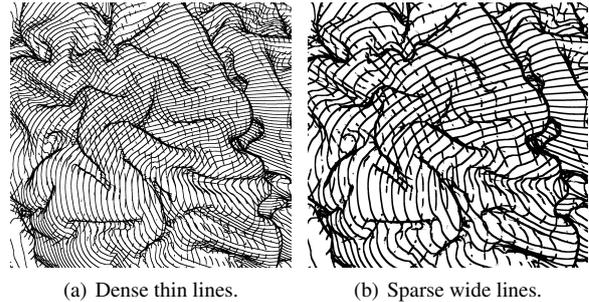


Figure 5: Precise line control.

lines (using $\frac{d}{2}$ as the distance between two cutting planes) as the user zooms in. However, we use only a single layer and simply adjust d continuously (Fig. 4) for changing zoom-levels. Even though this could lead to a "swimming" of the lines on the surface, this effect is not noticeable in practice since only single lines are added at a time for the extent of the entire dataset so that all existing lines only move a small percentage of the dataset's extent.

This general technique lets us hatch any surface at interactive to real-time frame-rates without smoothness constraints, connectivity information, or curvature values. Thus, we are able to work with an unordered set of polygons (i. e., a triangle soup) and are still able to generate a hatching with precise control of the line attributes such as line thickness and density (Fig. 5). In addition, we can also do cross-hatching and can modify the computation of d_p to obtain more flexible hatching patterns based on other intersection surfaces. For instance, we used $d_p = |F_p - 0.5| \cdot P_n$; $F_p \in [0, 1]$ for many of our examples which has the effect of creating somewhat rounded cutting planes and works well for many cases.

3.2.3. Ambient Occlusion

Ambient occlusion [ZIK98, Lan02] has proven to aid the perception of depth and shape of objects. Hence, we employ it to guide the decision of where to draw hatching lines and where to omit them, as motivated in Section 3.1. Specifically, we use screen-space ambient occlusion (SSAO) [SA07] in a separate pass to render the ambient occlusion values into an 8 bit G-buffer. The color resolution of 8 bit is sufficient

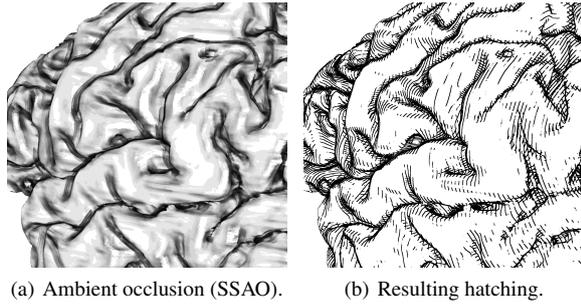


Figure 6: Use of ambient occlusion to control the hatching; SSAO image in (a) enhanced for illustration purposes.

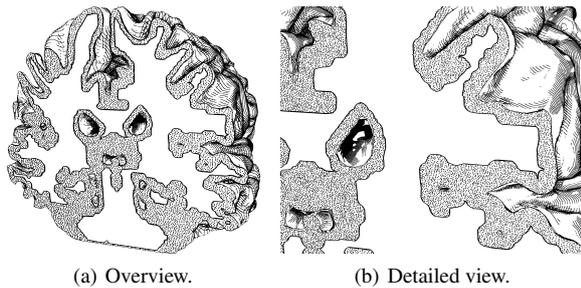


Figure 7: Use of stippling to illustrate gray matter.

for our purpose and the spatial resolution of the ambient occlusion G-buffer can even be half that of the regular frame buffer. Also, because SSAO often produces noisy images and banding artifacts (Fig. 6(a)), it is typically subjected to a Gaussian blur filter. We found, however, that these measures are not necessary in our case because these artifacts are not bothersome in combination with our hatching method.

We employ SSAO as the sole illumination technique and use it to control the spatial application of hatching as well as its parametrization (Fig. 6). Regions that are well illuminated do not receive any hatching lines while darker regions are hatched. In addition, we add additional layers of (cross-hatching) strokes for increasingly dark regions in the SSAO. Finally, for very dark regions we can also non-linearly increase the line width of the hatching strokes so that very dark areas in the SSAO may appear completely black in the hatching. These two effects increase the contrast and dynamic range of the illustration, an effect that can often be observed in hand-made illustration examples (e. g., Fig. 2(c)).

3.3. Cutting Planes and Gray Matter Stippling

We use cutting planes to visualize the inside of the brain and show the internal fiber tract structures, as in the examples in Fig. 2 and similar to previous approaches (e. g., [JBB*08]). However, to generate an illustrative visualization that is consistent with the pen-and-ink style of our examples we render the gray matter regions using stippling and leave the remaining intersection region white (Fig. 7). For this purpose we

(1) derive the intersection region, (2) determine the parts of it that represent gray matter, and (3) render this region with a fast and zoom-dependent stippling technique.

For the first step we intersect the cutting plane with the iso-surface we derived in Section 3.2.1. We render the cutting plane as a large quad and for each fragment p we check whether it is located inside the iso-surface by comparing its iso-value with the threshold. If this is the case we render it in white, otherwise we discard the fragment. To identify the gray matter regions (Step 2) we rely on segmentation data that can be derived similarly to the approach in [JBB*08].

For the final stippling step we employ the Wang tile technique presented by Kopf et al. [KCODL06]. We treat each cutting plane separately, place a virtual plane with the tiling into the scene, and use the Wang tile hierarchy to determine (on the CPU) which tiles are currently in view. We use an analytic representation of the stipples on a tile and place them using their hierarchy until we achieve the desired gray value. This ensures that the stipple points do not jump, but instead new ones are added as zooming occurs. In practice, we render the stipples as OpenGL points and filter them on the GPU with a stencil buffer that stores the extent of the gray matter region. One thing to note is that this approach considers stipple placement only in 2D and does not introduce perspective distortion of the stipple placement or shape. It is, therefore, similar to the hand-drawn stippling of gray matter in our examples (Fig. 2) where the cutting plane with the stippling is always perpendicular to the viewing direction.

3.4. Integration of Detail and Context

The final step is to integrate the three parts of our visualization (refer to the schematic pipeline in Fig. 8 and the example in Fig. 9): the (potentially partially cut) hatched surface of the brain and/or skull (Fig. 9(a)), the surface of the cutting planes with the stippled gray matter, and the DTI fiber tract visualization generated using the technique by Everts et al. [EBRI09] (Fig. 9(c)). For this purpose we render the fibers and the cutting plane mask separately into different frame buffers. The cutting plane mask consists of a white part and a gray region for the gray matter per cutting plane (Fig. 9(b)). The gray region uses different degrees of gray (color IDs) for the different cutting planes used in the rendering to be able to distinguish them from the stippling. Next, we compose the hatching and fiber tract layers and add the white cutting plane. The final pass then adds stipples for the gray matter only where stipples need to be placed according to the mask (Fig. 9(b)). Normally, we perform this compositing using a depth-sorted blending scheme, ensuring that hidden structures remain properly hidden. Alternatively, we also explored rendering the fiber tracts without a depth test, this allows us to look beyond the surface (Fig. 10).

In both cases we add an additional halo around the fiber tracts (Fig. 10) to visually separate them as focus objects and,

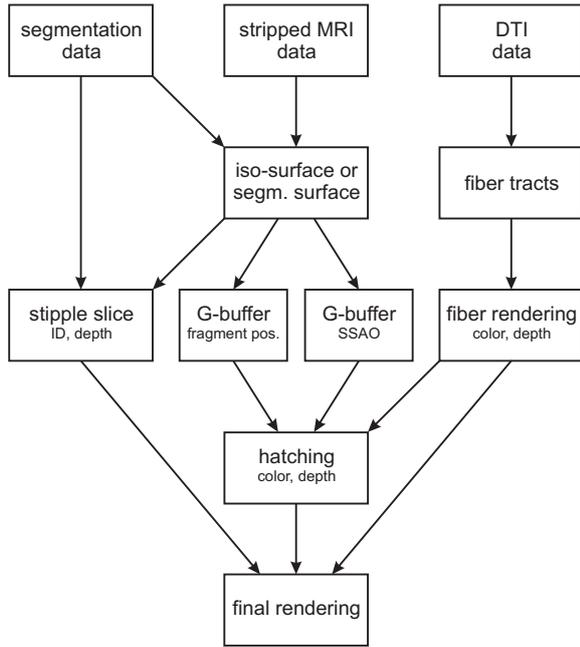


Figure 8: The data pipeline employed in our approach.

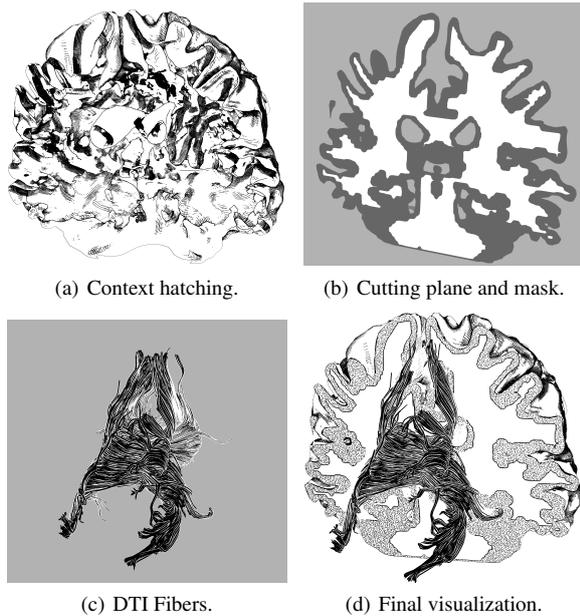
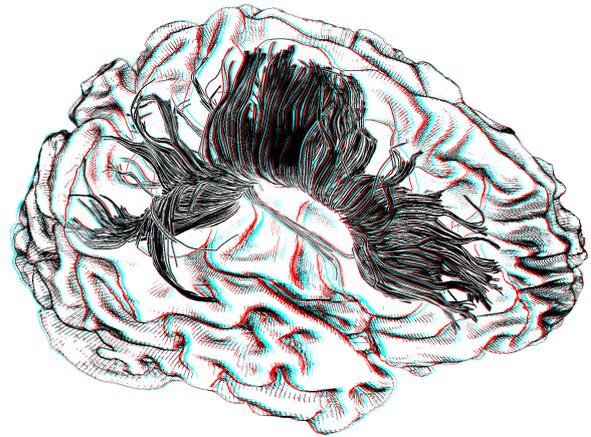


Figure 9: Compositing of the individual parts.

thus, emphasize them. This halo functionally differs from halos as used for the fiber tracts where they assist the perception of the spatial ordering of lines or line bundles. We determine the halo effect by looking at a 5×5 sampling grid (the maximum extent of this grid determines the width of the halo) in the neighborhood of every pixel that belongs to the surface hatching and compute the fraction of pixels that



(a) Regular 2D rendering.



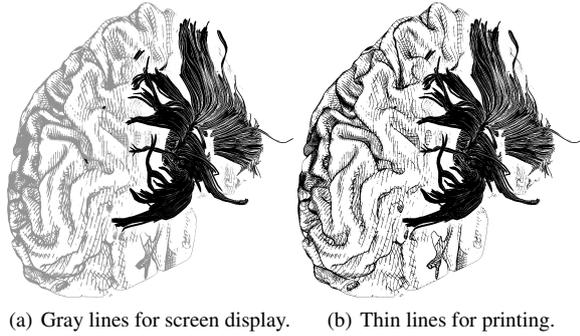
(b) Anaglyphic 3D rendering.

Figure 10: Compositing illustration with halo. Notice that the fiber tracts are actually located inside the brain's surface, an effect that is better visible in animations or in the anaglyphic 3D image (b).

represent fibers. This percentage is used to thin the lines of the hatching and/or making them brighter. Thus, this computation and, consequently, the fiber tract rendering is done before rendering the hatching. The thinning has the effect of a continuous transition between surface hatching and fiber tracts without the need to use transparency.

3.5. De-Emphasis of the Context

In addition to adding a larger halo around the fibers (Fig. 10), we also explored further means to visually emphasize the focus objects. This is necessary because otherwise the focus and the context may appear similarly important. For this purpose we control the attributes of the marks of the hatching and stippling, depending on the intended reproduction. For on-screen use we change the saturation of the marks to gray (Fig. 11(a)). This retains the same amount of detail



(a) Gray lines for screen display. (b) Thin lines for printing.

Figure 11: De-emphasis of the context to visually emphasize the fibers in the focus, depending on type of reproduction.

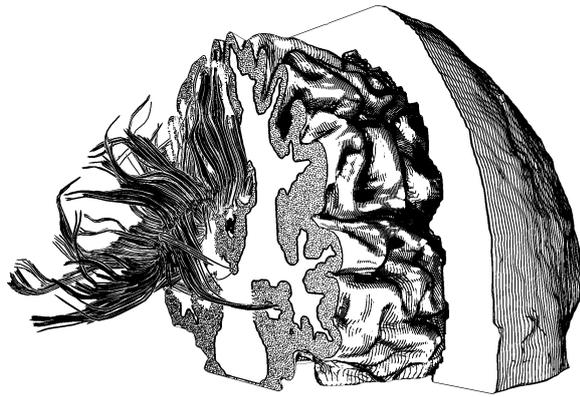


Figure 12: Example visualization to show fibers in the context of the brain and skull using cutting planes and a halo.

and makes use of the high color resolution without being (further) limited by the low pixel resolution. In contrast, for print reproduction we use thinner lines and smaller stipples (Fig. 11(b)), making use of the higher spatial resolution of print reproduction. Because our technique relies largely on GPU (pixel) rendering we do not generate vector output but instead produce a high-resolution black-and-white pixel image which achieves similar quality as vector rendering.

4. Results

In the following we first present two case studies that make use of the presented techniques and show how they can be used to illustrate DTI fiber tracts and its context in situ. Afterward, we give some details on the performance of the method and discuss a number of its limitations.

4.1. Fibers, Brain, and Skull

The compositing of DTI fiber tracts and context hatching permits a simultaneous illustrative display of both structures. This approach works well when only part of the context is shown, e. g., only one half of the brain as in Fig. 11 (in this

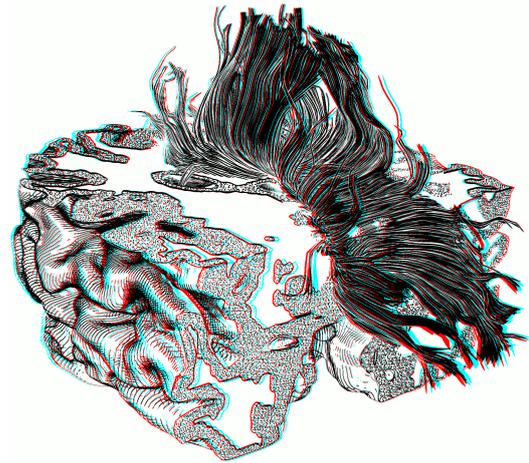


Figure 13: Anaglyphic rendering of the illustration in Fig. 1 with two cutting planes and a slight halo around the fibers.

and the following examples we use a manual selection of the fiber tracts extracted from the DTI data). The combination with a halo around the fiber tracts also makes it possible to show fiber tracts inside the brain without the use of cutting planes (Fig. 10). This effect is similar to the ones explored by Viola et al. [VKG05] and works best when animated or in stereoscopic rendering (e. g., anaglyphs, see Fig. 10(b)). Alternatively, cutting planes also allow users to look inside the brain as done in Figures 1, 12, and 13. Fig. 12 demonstrates both the use of iso-surface extraction for the brain with cross-hatching and surfaces from segmentation using parallel hatching. It is also possible to use more cutting planes as illustrated in Fig. 1 and the 3D version in Fig. 13.

4.2. Structures Inside the Brain

To illustrate the case of showing context for fiber visualization using objects inside the brain, Fig. 14 shows an image where a segmented tumor is rendered using hatching and a selection of fibers wraps around it. For the hatching of brain and tumor we used different settings, in particular with respect to the hatching frequency and thickness.

4.3. Performance and Limitations

We implemented our approach using C++, OpenGL, and GLSL. Our first dataset (used for the performance measuring and most images in the paper) consists of skull-stripped MRI data ($128 \times 128 \times 61$ voxels, T1-weighted) with matching DTI data (3,363 fiber tracts from 61 volumes of $128 \times 128 \times 61$ voxels) and segmentation data. The second dataset (Fig. 14) consist of MRI data ($512 \times 512 \times 176$ voxels, T1 post contrast, N3 intensity corrected) with matching DTI data (9,008 fiber tracts from 62 volumes of $128 \times 128 \times 72$ voxels) and segmentation data. For comparison, we give performance data for generating selected il-

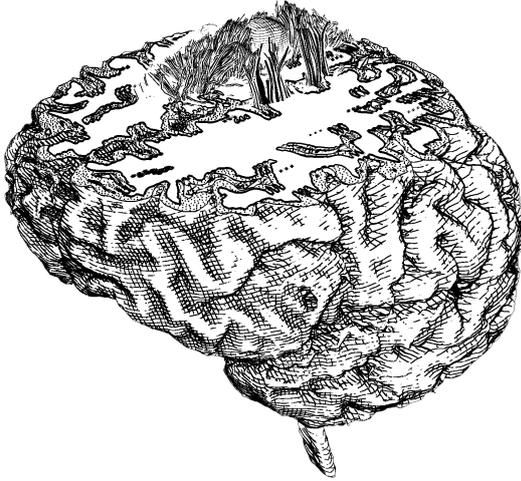


Figure 14: Fibers wrapping around a tumor. The segmentation data for the tumor was subdivided and smoothed for a better visualization; the gray matter area was approximated due to a lack of segmentation information in the dataset.

Table 1: Measurements of rendering performance (in fps); 2D—regular image, 3D—anaglyphic image, AA—with anti-aliasing, OH—only hatching (i. e., no fibers, no stippling).

Method/Figure	1/13	9	10	12	14
2D	33	34	24	29	24
2D-AA	19	19	13	16	15
3D	16	16	13	15	13
3D-AA	9	9	7	8	8
2D-OH	36	36	22	33	34
2D-AA-OH	21	20	12	16	19
3D-OH	20	20	11	16	18
3D-AA-OH	10	10	6	8	10

illustrations as shown in this paper on an Intel® Core™ i7 920 machine (64 bit, 2.66 GHz) with an NVIDIA® Quadro® FX 4800 under Windows® 7. Using a rendering size of 935 × 888 pixels, we measured the performance for both regular and anaglyphic rendering, for both without and with anti-aliasing (2× super-sampling), and also for the hatching technique only. The results are given in Table 1 and show that the pure hatching can achieve interactive to real-time frame-rates. It is interesting to note that the hatching does not depend on the number of hatching lines used or their parametrization, but that the program is fragment-bound. Thus the rendering times, also for the combined technique, depend mostly on the size of the image on the screen. Rendering high-quality pixel images takes not more than a second or two, for which we render the image to a large off-screen buffer (e. g., 2048 × 2048 pixels) and then write it to a file (which takes the most time).

One of the limitations of the technique lies in the necessity of specially prepared datasets to use its full potential.

While it is possible to directly use a MRI dataset with matching DTI data from which fiber tracts have been extracted, it is better to use additional segmentation data so that the MRI volume can be skull-stripped, that the skull can be dedicatedly rendered, and that the gray matter region can be illustrated medically correct. For all of these data preparation tasks special tools exist (e. g., SPM [FHW*95] or the FSL tool with the FAST package) so that, in fact, patient-specific data can be used for visualization. Another problem is that the halo that we use around the fiber tracts (focus) to set them apart from their surrounding context is sometimes difficult to notice and may be difficult to interpret in black-and-white still images (e. g., Fig. 10(a)). However, this problems does not occur for animations and 3D (anaglyphic) rendering (e. g., Fig. 10(b)). One final limitation that we want to mention is that we currently use simple techniques for line and stipple dot rendering. The use of techniques that are example-based (e. g., [KMI*09]) may be interesting to explore as well as the use of a stable noise function (e. g., [Per85]) for slightly perturbing the path of the hatching lines.

5. Informal Evaluation

The motivation for this work arose from discussions with a neuroscientist whose input was also used at several stages of the development. To be able to provide some evidence for the effectiveness and usefulness of the approach we invited him and one additional neuroscientist/anatomist for an informal evaluation to discuss the results. During this session we first demonstrated our program and then let the two experts explore the dataset interactively themselves. We also showed them printed examples of our illustrations.

Overall, they were very impressed by the illustrative visualizations, noting that they cannot create similar images with their current set of tools. Comments included “this is fantastic,” “elegant,” and “very nice.” In particular, they liked not only the aesthetics but also the precision and high detail of the depiction and said that the combination of fibers and halos helps them understand the spatial relationships. For example, they especially liked the image shown in Fig. 10 and, in general, anaglyphic 3D visualizations. Also, they said they could imagine that pathologic cases would be easier to spot with the precision the tool provides. In addition, they mentioned that it is essential to visualize major internal structures of the brain, an aspect that our approach solves only to some degree—future versions of our tool could use additional segmentations to support such functionality. As additions they asked for the ability to superimpose the MRI data slices on the cutting planes for comparison as well as the possibility to explore additional data such as probabilistic tractography data, brain activation data, or a grid for measurements.

One aspect they were particularly excited about and that they want to follow up on with us is to explore the use of the presented types of illustrations for publications of their own work. They reported on the problem of expensive additional

color charges which could be avoided with illustrations in pen-and-ink style. For this purpose they suggested and will help us to explore the possibility for an easy means of exporting the necessary data using methods such as SPM to be able to directly use it in our tool. They also see a potential for our approach to be applied in teaching about anatomy.

6. Conclusion

In summary, we have presented how a number of previously developed non-photorealistic rendering and visualization techniques can be combined and/or adapted to allow us to illustratively visualize DTI fiber tracts together with their context. This approach is inspired by traditional hand-drawn pen-and-ink illustration and is aimed at visualizing the data interactively but is also well suited to produce high-quality illustrations for print reproduction. We believe that the use of interactive slice-based hatching gives a good indication of the shape of the context objects. Also, the control of where to apply the hatching through screen-space ambient occlusion not only is able to generate darker regions for sulci but also generates darker rims as seen in hand-drawn examples.

In developing our approach we were inspired by discussions with a domain expert and have included his feedback throughout the process. In addition, we informally evaluated our results with him and another expert. This evaluation gave evidence for the usefulness of the technique, demonstrating that domain experts were excited about the precision and detail of the depiction. It also provided a range of paths for future work, including the exploration of using the approach in their day-to-day practice (possibly combined with [SSA*08]) and the publication of their results. While our work focused on black and white pen-and-ink illustration, it would also be straightforward in this context to use color to indicate the local orientation of fiber tracts as done in many traditional fiber tract visualizations.

We concentrated in this paper on the medical domain and, in particular, on brain data derived from MRI scans, but we believe that similar techniques can also be used for other application domains. Within the medical field, for example, we would like to explore the visualization of heart muscle fibers in the future. We could also envision applications in domains such as particle simulations in physics where the context for flows also needs to be shown. Additional future work includes, beyond addressing the limitations mentioned in Section 4.3, to work on interaction techniques that allow an intuitive combination of view selection, cutting plane placement, and specification of a fiber subset for visualization.

Acknowledgments

We thank Leonardo Cerliani from the Neuroimaging Center, Univ. Groningen, for the first dataset; the second dataset (Fig. 14) is courtesy of Prof. B. Terwey, Klinikum Mitte, Bremen, Germany. We also thank our colleagues and the

study participants for interesting discussions. This research is partially funded by the Dutch National Science Foundation (NWO), “VIEW” project, project no. 643.100.501.

References

- [ARS79] APPEL A., ROHLF F. J., STEIN A. J.: The Haloed Line Effect for Hidden Line Elimination. *ACM SIGGRAPH Computer Graphics* 13, 3 (Aug. 1979), 151–157.
- [BG07] BRUCKNER S., GRÖLLER E.: Enhancing Depth-Perception with Flexible Volumetric Halos. *IEEE Transactions on Visualization and Computer Graphics* 13, 6 (Nov./Dec. 2007), 1344–1351.
- [BJH*09] BORN S., JAINEK W. M., HLAWITSCHKA M., TRANTAKIS C., MEIXENSBERGER J., BARTZ D.: Multimodal Visualization of Dti and fMRI Data Using Illustrative Methods. In *Bildverarbeitung für die Medizin* (2009), Springer, Berlin, Heidelberg, pp. 6–10.
- [BKR*05] BURNS M., KLAWE J., RUSINKIEWICZ S., FINKELSTEIN A., DECARLO D.: Line Drawings from Volume Data. *ACM Transactions on Graphics* 24, 3 (Aug. 2005), 512–518.
- [CTdS08] CATANI M., THIEBAUT DE SCHOTTEN M.: A Diffusion Tensor Imaging Tractography Atlas for Virtual in Vivo Dissections. *Cortex* 44, 8 (Sept. 2008), 1105–1132.
- [Dau05] DAUBER W.: *Feneis' Bild-Lexikon der Anatomie*, 9th ed. Georg Thieme Verlag, Stuttgart, 2005.
- [DCLK03] DONG F., CLAPWORTHY G. J., LIN H., KROKOS M. A.: Nonphotorealistic Rendering of Medical Volume Data. *IEEE Computer Graphics & Applications* 23, 4 (July/Aug. 2003), 44–52.
- [DHR*99] DEUSSEN O., HAMEL J., RAAB A., SCHLECHTWEIG S., STROTHOTTE T.: An Illustration Technique Using Intersections and Skeletons. In *Proc. Graphics Interface* (1999), Morgan Kaufmann, San Francisco, pp. 175–182.
- [DWS*88] DEERING M., WINNER S., SCHEDIWIY B., DUFFY C., HUNT N.: The Triangle Processor and Normal Vector Shader: A VLSI System for High Performance Graphics. *ACM SIGGRAPH Computer Graphics* 22, 4 (Aug. 1988), 21–30.
- [EBRI09] EVERTS M. H., BEKKER H., ROERDINK J. B. T. M., ISENBERG T.: Depth-Dependent Halos: Illustrative Rendering of Dense Line Data. *IEEE Transactions on Visualization and Computer Graphics* 15, 6 (Nov./Dec. 2009), 1299–1306.
- [Elb95] ELBER G.: Line Illustrations ∈ Computer Graphics. *The Visual Computer* 11, 6 (June 1995), 290–296.
- [ER00] EBERT D., RHEINGANS P.: Volume Illustration: Non-Photorealistic Rendering of Volume Models. In *Proc. Visualization* (2000), IEEE Computer Society, Los Alamitos, pp. 195–202.
- [ES06] EBERT D. S., SOUSA M. C. (Eds.): *Illustrative Visualization for Medicine and Science* (2006), vol. 6 of *ACM SIGGRAPH 2006 Course Notes*, ACM SIGGRAPH.
- [FHW*95] FRISTON K. J., HOLMES A., WORSLEY K., POLINE J.-B., FRITH C., FRACKOWIAK R.: Statistical Parametric Maps in Functional Imaging: A General Linear Approach. *Human Brain Mapping* 2, 4 (1995), 189–210.
- [FMS01] FREUDENBERG B., MASUCH M., STROTHOTTE T.: Walk-Through Illustrations: Frame-Coherent Pen-and-Ink Style in a Game Engine. *Computer Graphics Forum* 20, 3 (Sept. 2001), 184–191.
- [GIHL00] GIRSHICK A., INTERRANTE V., HAKER S., LEMOINE T.: Line Direction Matters: An Argument for the Use of Principal Directions in 3D Line Drawings. In *Proc. NPAR* (2000), ACM, New York, pp. 43–52.

- [HH04] HARGREAVES S., HARRIS M.: *Deferred Shading*. Whitepaper and presentation, NVIDIA, 2004.
- [Hod03] HODGES E. R. S. (Ed.): *The Guild Handbook of Scientific Illustration*, 2nd ed. John Wiley & Sons, Hoboken, NJ, 2003.
- [HP60] HOUSE E. L., PANSKY B.: *A Functional Approach to Neuroanatomy*. McGraw-Hill Book Company, New York, 1960.
- [HZ00] HERTZMANN A., ZORIN D.: Illustrating Smooth Surfaces. In *Proc. SIGGRAPH (2000)*, ACM, New York, pp. 517–526.
- [IFP97] INTERRANTE V., FUCHS H., PIZER S. M.: Conveying the 3D Shape of Smoothly Curving Transparent Surfaces via Texture. *IEEE Transactions on Visualization and Computer Graphics* 3, 2 (Apr.–June 1997), 98–117.
- [JBB*08] JAINEK W. M., BORN S., BARTZ D., STRASSER W., FISCHER J.: Illustrative Hybrid Visualization and Exploration of Anatomical and Functional Brain Data. *Computer Graphics Forum* 27, 3 (May 2008), 855–862.
- [KCODL06] KOPF J., COHEN-OR D., DEUSSEN O., LISCHINSKI D.: Recursive Wang Tiles for Real-Time Blue Noise. *ACM Transactions on Graphics* 25, 3 (July 2006), 509–518.
- [KMI*09] KIM S., MACIEJEWSKI R., ISENBERG T., ANDREWS W. M., CHEN W., SOUSA M. C., EBERT D. S.: Stippling By Example. In *Proc. NPAR (2009)*, ACM, New York, pp. 41–50.
- [Lan02] LANDIS H.: Production-Ready Global Illumination. In *Course Notes of SIGGRAPH 2002*, no. 16. ACM, New York, 2002, ch. 5, pp. 331–338.
- [Lei94] LEISTER W.: Computer Generated Copper Plates. *Computer Graphics Forum* 13, 1 (Mar. 1994), 69–77.
- [LMT*03] LU A., MORRIS C. J., TAYLOR J., EBERT D. S., HANSEN C., RHEINGANS P., HARTNER M.: Illustrative Interactive Stipple Rendering. *IEEE Transactions on Visualization and Computer Graphics* 9, 2 (Apr.–June 2003), 127–138.
- [MvZ02] MORI S., VAN ZIJL P. C.: Fiber Tracking: Principles and Strategies – A Technical Review. *NMR Biomed* 15, 7–8 (Nov./Dec. 2002), 468–480.
- [NGE*06] NIMSKY C., GANSLAND O., ENDERS F., MERHOF D., HAMMEN T., BUCHFELDER M.: Visualization Strategies for Major White Matter Tracts for Intraoperative use. *International Journal of Computer Assisted Radiology and Surgery* 1, 1 (Mar. 2006), 13–22.
- [NSW02] NAGY Z., SCHNEIDER J., WESTERMANN R.: Interactive Volume Illustration. In *Proc. Vision, Modeling and Visualization (2002)*, Akademische Verlagsgesellschaft Aka GmbH, Berlin, pp. 497–504.
- [Ost99] OSTROMOUKHOV V.: Digital Facial Engraving. In *Proc. SIGGRAPH (1999)*, ACM, New York, pp. 417–424.
- [Per85] PERLIN K.: An Image Synthesizer. *SIGGRAPH Computer Graphics* 19, 3 (July 1985), 287–296.
- [PHWF01] PRAUN E., HOPPE H., WEBB M., FINKELSTEIN A.: Real-Time Hatching. In *Proc. SIGGRAPH (2001)*, ACM, New York, pp. 581–586.
- [RHD*06] RITTER F., HANSEN C., DICKEN V., KONRAD O., PREIM B., PEITGEN H.-O.: Real-Time Illustration of Vascular Structures. *IEEE Transactions on Visualization and Computer Graphics* 12, 5 (Sept./Oct. 2006), 877–884.
- [SA07] SHANMUGAM P., ARIKAN O.: Hardware Accelerated Ambient Occlusion Techniques on GPUs. In *Proc. I3D (2007)*, ACM, New York, pp. 73–80.
- [SALS96] SALISBURY M. P., ANDERSON C., LISCHINSKI D., SALESIN D. H.: Scale-Dependent Reproduction of Pen-and-Ink Illustration. In *Proc. SIGGRAPH (1996)*, ACM, New York, pp. 461–468.
- [SFWS03] SOUSA M. C., FOSTER K., WYVILL B., SAMAVATI F.: Precise Ink Drawing of 3D Models. *Computer Graphics Forum* 22, 3 (Sept. 2003), 369–379.
- [SPW*07] SCHMAHMANN J. D., PANDYA D. N., WANG R., DAI G., D’ARCEUIL H. E., DE CRESPIGNY A. J., WEDEEN V. J.: Association Fibre Pathways of the Brain: Parallel Observations from Diffusion Spectrum Imaging and Autoradiography. *Brain* 130, 3 (Mar. 2007), 630–653.
- [SSA*08] SCHULTZ T., SAUBER N., ANWANDER A., THEISEL H., SEIDEL H.-P.: Virtual Klingler Dissection: Putting Fibers into Context. *Computer Graphics Forum* 27, 3 (May 2008), 1063–1070.
- [ST90] SAITO T., TAKAHASHI T.: Comprehensible Rendering of 3-D Shapes. *ACM SIGGRAPH Computer Graphics* 24, 3 (Aug. 1990), 197–206.
- [TC00] TREAVENTT S. M. F., CHEN M.: Pen-and-Ink Rendering in Volume Visualisation. In *Proc. Visualization (2000)*, IEEE Computer Society, Los Alamitos, CA, USA, pp. 203–210.
- [TCM06] TARINI M., CIGNONI P., MONTANI C.: Ambient Occlusion and Edge Cueing for Enhancing Real Time Molecular Visualization. *IEEE Transactions on Visualization and Computer Graphics* 12, 5 (Sept./Oct. 2006), 1237–1244.
- [TIP05] TIETJEN C., ISENBERG T., PREIM B.: Combining Silhouettes, Shading, and Volume Rendering for Surgery Education and Planning. In *Proc. EuroVis (2005)*, Eurographics Association, Aire-la-Ville, Switzerland, pp. 303–310.
- [TSD07] TATARCHUK N., SHOPF J., DECORO C.: Real-Time Isosurface Extraction Using the GPU Programmable Geometry Pipeline. In *ACM SIGGRAPH 2007 Courses*, no. 28. ACM, New York, 2007, ch. 9, pp. 122–137.
- [VGH*05] VIOLA I., GRÖLLER M. E., HADWIGER M., BUHLER K., PREIM B., COSTA SOUSA M., EBERT D., STREDNEY D.: Illustrative Visualization. In *Course Notes of IEEE VIS 2005*. IEEE Computer Society, Los Alamitos, 2005.
- [VKG05] VIOLA I., KANITSAR A., GRÖLLER M. E.: Importance-Driven Feature Enhancement in Volume Visualization. *IEEE Transactions on Visualization and Computer Graphics* 11, 4 (July/Aug. 2005), 408–418.
- [WJNP*04] WAKANA S., JIANG H., NAGAE-POETSCHER L. M., VAN ZIJL P. C. M., MORI S.: Fiber Tract-based Atlas of Human White Matter Anatomy. *Radiology* 230, 1 (Jan. 2004), 77–87.
- [WS94] WINKENBACH G. A., SALESIN D. H.: Computer-Generated Pen-and-Ink Illustration. In *Proc. SIGGRAPH (1994)*, ACM, New York, pp. 91–100.
- [YC04] YUAN X., CHEN B.: Illustrating Surfaces in Volume. In *Proc. VisSym (2004)*, Eurographics Association, Aire-la-Ville, Switzerland, pp. 9–16, 337.
- [ZDL03] ZHANG S., DEMIRALP Ç., LAIDLAW D. H.: Visualizing Diffusion Tensor MR Images Using Streamtubes and Stream-surfaces. *IEEE Transactions on Visualization and Computer Graphics* 9, 4 (Oct.–Dec. 2003), 454–462.
- [ZIK98] ZHUKOV S., INOES A., KRONIN G.: An Ambient Light Illumination Model. In *Rendering Techniques (1998)*, Springer-Verlag, Wien, New York, pp. 45–56.
- [ZISS04] ZANDER J., ISENBERG T., SCHLECHTWEIG S., STROTHOTTE T.: High Quality Hatching. *Computer Graphics Forum* 23, 3 (Sept. 2004), 421–430.