A Constraint-Based Technique for Haptic Volume Exploration

Milan Ikits J. Dean Brederson Charles D. Hansen Christopher R. Johnson

Scientific Computing and Imaging Institute, University of Utah*

Abstract

We present a haptic rendering technique that uses directional constraints to facilitate enhanced exploration modes for volumetric datasets. The algorithm restricts user motion in certain directions by incrementally moving a proxy point along the axes of a local reference frame. Reaction forces are generated by a spring coupler between the proxy and the data probe, which can be tuned to the capabilities of the haptic interface. Secondary haptic effects including field forces, friction, and texture can be easily incorporated to convey information about additional characteristics of the data. We illustrate the technique with two examples: displaying fiber orientation in heart muscle layers and exploring diffusion tensor fiber tracts in brain white matter tissue. Initial evaluation of the approach indicates that haptic constraints provide an intuitive means for displaying directional information in volume data.

CR Categories: I.3.7 [Computer Graphics]: Three-Dimensional Graphics and Realism—Virtual Reality; D.2.2 [Software Engineering]: Design Tools and Techniques—User Interfaces; H.5.2 [Information Systems]: User Interfaces—Haptic I/O

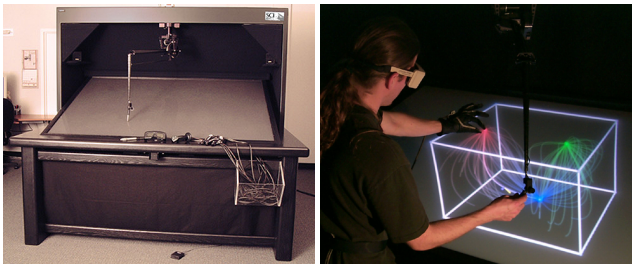
Keywords: haptic rendering, immersive visualization, human-computer interaction

1 Introduction

A primary advantage of haptic rendering is that it provides a bidirectional flow of information via position sensing and force feedback. This coupled information flow results in more natural and intuitive interaction and utilizes additional sensory channel bandwidth of the user. When users are presented with a proper combination of visual and haptic information, they experience a sensory synergy resulting from physiological reinforcement of the displayed multimodal cues [Durlach and Mavor 1994].

Implementations of the traditional visualization pipeline typically provide a limited set of interactive data exploration capabilities. Tasks such as finding and measuring features in the data or investigating the relationship between different quantities may be easier to perform with more natural data exploration tools. To develop visualization and exploration techniques that further increase insight and intuitive understanding of scientific datasets, we designed and built an integrated immersive visual and haptic system, the Visual Haptic Workbench [Brederson et al. 2000] (Figure 1(a)).

*{ikits|jdb|hansen|crj}@sci.utah.edu



(a) (b)

Figure 1: (a) The Visual Haptic Workbench. (b) A user explores a volumetric vector dataset. The data probe is constrained along a streamline resulting in intuitive haptic feedback.

Haptic feedback has been shown advantageous as an interaction modality for specific tasks [Wall and Harwin 2000; Basdogan et al. 1998]. For visualization purposes, a rich lexicon of kinesthetic and tactile cues can be displayed to represent salient features of the data. For example, surfaces can be rendered with curvature information overlaid or edges enhanced to highlight discontinuities, transitions, and homogeneous regions. They can be dynamic or deformable, augmented with friction, texture, and realistic or synthetic surface material properties. It is also possible to fortify the user interface with haptic guides to further aid the user in the exploration task.

In this paper we present a haptic rendering technique that uses directional constraints to provide intuitive exploration modes for volumetric datasets. Haptic constraints are implemented by incrementally moving a proxy point along the axes of a local reference frame. The motion of the proxy is controlled by simple rules as well as transfer functions that provide a straightforward framework for incorporating various haptic effects into the data probing task.

For example, to guide the user in vector field data, the proxy can be constrained along a streamline such that any effort to move the probe in a direction perpendicular to the current orientation of the field results in a strong opposing force. However, if the user pushes the probe hard enough, the proxy could “pop over” to an adjacent streamline, allowing the user to move the probe in three dimensions and still receive strong haptic cues about the orientation of the flow. We can use an additional force component along the streamline to indicate the magnitude of the field. Alternatively, a secondary constraint can be added to convey information about the speed of the flow in the form of haptic “tickmarks”. We found that such techniques result in intuitive feedback when exploring vector field data (Figure 1(b)). A recent study on the effectiveness of various haptic rendering techniques for CFD datasets reached a similar conclusion [van Reimersdahl et al. 2003].

After reviewing the general classes of haptic rendering methods from the literature and discussing the motivation for our approach, we present the implementation details of constraint-based haptic data rendering. We illustrate our technique with two examples: displaying muscle fiber orientation in a heart model and exploring white matter fiber tract connectivity in brain diffusion tensor MRI data.

2 Background and Motivation

To motivate the constraint-based method described in the following section, we first discuss the evolution of haptic rendering methods from the literature. We divide haptic rendering techniques into two categories: surface methods and volumetric methods.

2.1 Surface Rendering

The majority of haptic rendering algorithms are geometric in nature, since they deal with the problem of interacting with various surface representations. Surface rendering requires a suitable geometric model, a rapid collision detection technique, an incremental surface tracing algorithm, and a model for generating contact forces from the probe-surface interaction. Surface tracing algorithms exist for a variety of representations, including polygonal, parametric (NURBS), and implicit surfaces. These algorithms employ a combination of global and local distance queries to track the geometry closest to the interaction point. Haptic surface rendering has evolved from simple force-field methods [Massie 1993] to constraint-based approaches utilizing a proxy point [Zilles and Salisbury 1995; Ruspini et al. 1997]. More recently, efficient techniques have emerged for haptic display of contact between complex polygonal objects [Kim et al. 2002; Otaduy and Lin 2003].

Contact forces are usually modeled by the probe interacting with a viscoelastic surface. A virtual spring and damper is used to mechanically couple the probe with the proxy during contact. From the visualization point of view, surfaces are represented by unilateral constraints that prevent the proxy from penetrating the object. Much of previous research has focused on improving the crispness of rendered surfaces as well as augmenting them with various material properties to create realistic and convincing virtual objects [Salisbury et al. 1995; Massie 1996; Srinivasan and Basdogan 1997].

2.2 Volume Rendering

Early work in haptic visualization used volumetric methods for exploring scalar and vector fields as well as molecular interactions [Brooks et al. 1990; Iwata and Noma 1993]. The majority of previous methods for haptic display of volume data properties are based upon a functional relationship between the reflected force and torque vectors as well as the probe state and local data measures:

$$\vec{F} = \vec{F}\,(X, D, T) \tag{1}$$

where X denotes the state, typically position $\vec{x}$ and velocity $\dot{\vec{x}}$ of the haptic probe, D represents a set of local data measures at the probe position, and T stands for a set of haptic transfer functions and rendering parameters. We borrow the term *force field rendering* for this class of techniques. The simplest examples in this category include density modulated viscous drag for scalar data [Avila and Sobierajski 1996; Pao and Lawrence 1998] and direct display of vector data [Iwata and Noma 1993; Mascarenhas et al. 2002]:

$$\vec{F}(\{\vec{x},\dot{\vec{x}}\},\{s(\vec{x})\},\{k(s)\}) = -k(s(\vec{x}))\,\dot{\vec{x}} \tag{2}$$

$$\vec{F}(\{\vec{x}\},\{\vec{v}(\vec{x})\},\{k\}) = k\,\vec{v}(\vec{x}) \tag{3}$$

where the gain k is adjusted according to the scale and magnitude of the data measures and the capabilities of the haptic interface. Note that in (2) we modulate viscous drag as a function of data value and in (3) we apply a force directly proportional to the local field vector.

Even though this approach represents an important step in the evolution of haptic data rendering techniques, it suffers from several limitations. First, it provides limited expressive power, because it is difficult to display and emphasize features in a purely functional form. For example, we found that using complex transfer functions for rendering isosurfaces is less convincing than traditional surface rendering approaches [Avila and Sobierajski 1996; Infed et al. 1999; Lawrence et al. 2000]. The reason for this is that the notion of *memory* is missing from these formulations [Salisbury and Tarr 1997; Lundin 2001]. Second, the device capabilities are captured implicitly in the rendering parameters. Applying a force as a function of probe state can easily result in instability, especially when several rendering modes are combined. It is very difficult and tedious to tune the behavior of the dynamical system formed by the force field equation (1) and the motion equations of the haptic device by finding an appropriate set of rendering parameters.

Fortunately, haptic rendering stability can be guaranteed when using a virtual coupling network [Colgate and Brown 1995; Adams and Hannaford 1998]. The coupler acts as a low-pass filter between the haptic display and the virtual environment, thereby limiting the maximum impedance that needs to be exhibited by the device and preventing the accumulation of energy in the coupled system [Renz et al. 2001; Hannaford and Ryu 2002]. Although the coupler is not part of the environment, the commonly used spring-damper form had been introduced implicitly in constraint-based surface rendering algorithms. In the next section we develop a similar approach to stable haptic display of directional information in volumetric datasets.

2.3 Haptic Constraints

Constraints have been used successfully in both telerobotics and haptics applications. In early work, virtual fixtures or guides were shown to improve operator performance in robot teleoperation tasks [Rosenberg 1993]. More recently, a haptic rendering framework was developed with algebraic constraints as the foundation [Hutchins 2000]. Constraints have been shown helpful in guiding the user in a goal-directed task [Gunn and Marando 1999]. User interfaces can also benefit from guidance. Examples include a haptic version of the common desktop metaphor [Miller and Zeleznik 1998] and a more natural paradigm for media control [Snibbe et al. 2001].

We found that constraints provide a useful and general foundation for developing haptic rendering algorithms for scientific datasets. For example, constrained spatial probing for seeding computational and visualization algorithms local to the proxy, *e.g.* particle advection, typically results in more cohesive insight than its unconstrained version. Volumetric constraints are obtained by augmenting the proxy with a local reference frame and controlling its motion according to a set of rules and transfer functions along the axes of the frame. This approach has the advantage that it provides a uniform basis for rendering a variety of data modalities. Thus, similar or closely related methods can be applied to seemingly unrelated datasets such that the result is a consistent interaction experience.

Algorithms for constrained point-based 3DOF haptic rendering have recently been developed for scalar density data [Blezek and Robb 1999; Lundin 2001] as well as vector fields used in computational fluid dynamics visualization and animation motion control applications [van Reimersdahl et al. 2003; Donald and Henle 2000]. Haptic constraints have also been successfully used for displaying molecular flexibility [Křenek 2000]. Techniques utilizing complex proxy geometry transform the proxy to a point shell to perform approximate 6DOF force and torque calculations using the individual point locations [McNeely et al. 1999; Renz et al. 2001; Mascarenhas et al. 2002; Petersik et al. 2002]. In the following sections we describe a general algorithm for 3DOF haptic data rendering using a constrained point-proxy and illustrate it with several examples.

3 Haptic Rendering with a Constrained Proxy Point

In general, haptic volume rendering algorithms based on a proxy point include four components that are executed at every iteration of the servo loop (see Figure 2):

1. Compute local data measures at current proxy location: Data values and other measures, *e.g.* gradient or curvature information are obtained from interpolating data elements around the current proxy location. The typical methods of choice are linear and trilinear interpolation, although higher order techniques may be more appropriate depending on the scale and resolution of the display [Sankaranarayanan et al. 2002]. Since haptic rendering is a local process just like particle advection, point-location algorithms for vector-field visualization on curvilinear and unstructured grids are readily applied [Novoselov et al. 2002]. A *local reference frame* $(\vec{e}_1, \vec{e}_2, \vec{e}_3)$ is a key component of constraint-based techniques. Examples include the frame defined by the gradient and principal curvature directions in scalar data and the frame of eigenvectors in diffusion tensor data. Note that the reference frame may be ill-defined or not exist. Thus, an important requirement for the algorithm is to compute a stable force response even when transitioning into and out of homogeneous and ill-posed regions in the data. For example, in scalar volumes the reference frame is poorly defined in regions where the gradient vanishes. One way to achieve smooth transitioning is to modulate the force output as a function of gradient magnitude [Lundin 2001]. Another example is the problem of specifying transfer functions such that isotropic regions are handled properly in diffusion tensor data. In this case the transfer function has to be constructed such that the force output either vanishes or degenerates to an isotropic point constraint.

2. Evaluate haptic transfer functions to determine rendering parameters: Similarly to graphical visualizations, the goal of haptic transfer functions is to emphasize and combine features in the data. For example, a transfer function can be used to specify apparent stiffness and friction for isosurface regions based on data value and gradient magnitude [Lundin 2001]. In contrast to visual transfer functions, haptic transfer function design is an unexplored area. Although it has been demonstrated that it is possible to faithfully reproduce measured material properties [Okamura et al. 1998], synthesizing them from different or abstract data remains a difficult problem. In this work we utilize stiffness and drag threshold transfer functions $\vec{\kappa}$ and $\vec{\tau}$ to constrain the motion of the proxy along the axes of the local reference frame.

3. Update proxy state: In this step the state of the proxy is updated according to simple motion rules. We have chosen a purely geometric approach, which updates the proxy location based on probe motion and rendering parameters along the axes of the local frame:

$$\vec{p}_k = \vec{p}_{k-1} + \Delta\vec{p} = \vec{p}_{k-1} + \sum_{i=1}^{3} \Delta p_i \, \vec{e}_i \tag{4}$$

where Δp_i is a function of probe position relative to the previous proxy location $\Delta x_i = (\vec{x}_k - \vec{p}_{k-1}) \cdot \vec{e}_i$. For example, surface haptics algorithms locally constrain the proxy to the tangent plane by setting the normal component of change to zero. More sophisticated strategies incorporate the force response from previous steps as well as other state variables. For example, physically based models assume the proxy has mass m and is moving in a medium with viscosity b [Snibbe et al. 1998]:

$$m\,\ddot{p}_i + b\,\dot{p}_i = F_i \tag{5}$$

where F_i is the force component acting on the proxy point along $\vec{e}_i$. Friction effects can be incorporated by adding and moving a static friction point within the constraint subspace [Salisbury et al. 1995].

Note that the linear approximation used in (4) is not always appropriate for expressing a nonlinear constraint such as staying on a surface or following a streamline. For example, when tracing volumetric isosurfaces, the first-order approximation obtained by projecting the probe point to the tangent plane defined by the gradient at the proxy location will result in the algorithm quickly loosing track of the surface. Thus, we find the new proxy location $\vec{p}_k$ by refining the initial estimate using Newton-Raphson iteration along the gradient direction [Salisbury and Tarr 1997]:

$$\Delta\vec{p} = -\frac{(s(\vec{p}) - s_0)\,\nabla s(\vec{p})}{|\,\nabla s(\vec{p})\,|^2} \tag{6}$$

where s_0 is the target isovalue. The refinement is terminated when the step size $|\Delta\vec{p}|$ is sufficiently small or when it reaches the maximum number of iterations permitted. Similarly, higher-order integration schemes, *e.g.* the fourth-order Runge-Kutta method, are necessary for computing the reference direction when following streamlines in vector data. For larger step sizes, supersampling and iterating steps 1–3 may be required to ensure constraints are satisfied accurately [Salisbury and Tarr 1997; Blezek and Robb 1999].

4. Compute force response: When using the spring-damper form of virtual coupling, the force response is computed from:

$$\vec{F}_k = k_c\,(\vec{x}_k - \vec{p}_k) - b_c\,(\dot{\vec{x}}_k - \dot{\vec{p}}_k) \tag{7}$$

where k_c and b_c are chosen according to the device capabilities. The optimal choice maximizes the coupling stiffness without causing instability [Adams and Hannaford 1998]. One problem is that these parameters may not be constant throughout the workspace. A particular choice that works well in the center may cause instability near the perimeter. Nevertheless, we can tune them by applying a point constraint at different locations in the workspace and determining which settings cause the device to become unstable by itself, *i.e.* without a stabilizing grasp. Analysis of the parameters could reveal the optimal operational region within the workspace of the device. In this paper we exclude the second term from (7), since filtering velocity is difficult without high resolution position measurements [Colgate and Brown 1995].

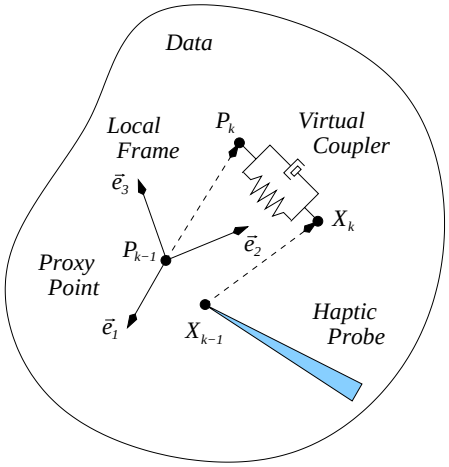
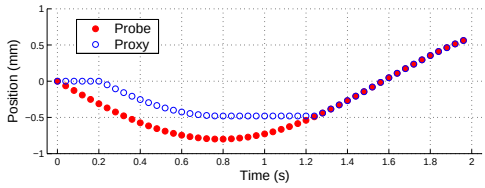
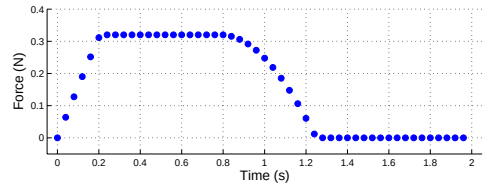
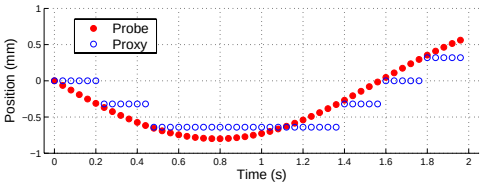


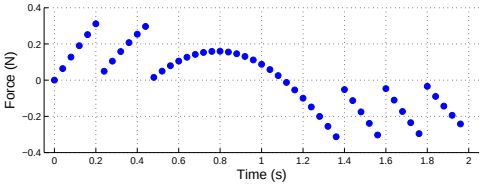
Figure 2: Components of constrained point-based 3DOF haptic data rendering. At time step k the state of the haptic probe has changed from X_{k-1} to X_k. The proxy state gets updated from P_{k-1} to P_k from which the force response is computed using a virtual coupler. Proxy update is based on data measures at the previous proxy location as well as haptic transfer functions and rendering parameters.





(a) Unilateral constraint.





(b) Snap-drag constraint.

Figure 3: One dimensional motion rule examples: (a) unilateral drag and (b) bilateral snap-drag. The motion of the probe and the proxy as a function of time is represented by the filled and empty circles, respectively. The resulting force responses are shown in the lower parts of the figures. Note that the sampling does not correspond to the haptic update rate.

4 Motion Rules and Transfer Functions

Motion rules allow us to create various haptic effects that we can further modulate using haptic transfer functions. One such effect simulates plastic material behavior by generating increasing resistance between the probe and the proxy until a certain threshold is reached. At this point the proxy is allowed to move towards the probe keeping the reaction force at the same level. This effect is expressed succinctly by the following formula:

$$\Delta p_i = \mathrm{sgn}(\Delta x_i)\,\max\left(|\Delta x_i| - \tau_i, 0\right) \tag{8}$$

This model yields free space motion when $\tau_i = 0$:

$$\Delta p_i = \Delta x_i \tag{9}$$

and a *bilateral* constraint when $\tau_i > 0$. We use the term *drag threshold* for τ_i, since it controls the difficulty of dragging the proxy along axis $\vec{e}_i$. Note that a stationary constraint is obtained when τ_i is sufficiently large, since it would take considerable effort to move the probe away from the proxy while resisting the increasing amount of force between them.

A *unilateral* constraint, which is the basis for surface rendering algorithms, is obtained by considering the direction of travel along the axis:

$$\Delta p_i = \begin{cases} \Delta x_i & \text{if} \quad \Delta x_i > 0 \\ \max\left(\Delta x_i - \tau_i, 0\right) & \text{if} \quad \Delta x_i \leq 0 \end{cases} \tag{10}$$

A bilateral *snap-drag* constraint changes the proxy location in discrete steps:

$$\Delta p_i = \begin{cases} \tau_i & \text{if} \quad \Delta x_i > \tau_i \\ 0 & \text{if} \quad \Delta x_i \leq \tau_i \end{cases} \tag{11}$$

The latter two rules are shown in Figure 3 along with the resulting force responses.

A different way to control proxy motion is achieved by scaling the force output according to stiffness transfer function $\vec{\kappa}$:

$$F_{k,i} = \kappa_i k_c \left(x_{k,i} - p_{k,i}\right) \tag{12}$$

where $0 \leq \kappa_i \leq 1$. This reduces the force required for dragging the proxy. Note that setting either τ_i or κ_i to zero produces no force output and creates frictionless motion along the axis. However, it yields two different proxy behaviors, since in the first case the proxy follows the probe exactly, while in the second case it lags behind by distance τ_i. Both parameters are necessary though, because we wish to express a range of effects from subtle directional hints to stiff rigid constraints. To illustrate why one parameter cannot be substituted for the other, consider following an elastic surface with varying stiffness. As illustrated in Figure 4, changing the stiffness of the coupler is equivalent to moving a copy of the proxy towards the probe [Ruspini et al. 1997]. It is not possible to achieve the same effect with the drag constraint alone, since the probe would have to move to the other side of the proxy to drag it back to the surface. Thus, it is necessary to maintain two separate but not completely independent transfer functions to create both elastic and plastic material behavior.

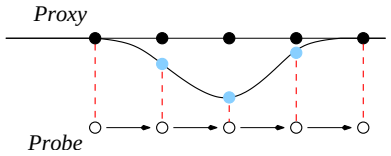


Figure 4: Changing the coupling stiffness or moving a copy of the proxy for force calculations are equivalent for achieving varying elastic constraints. In this example the stiffness is reduced as the proxy follows the probe resulting in a "deformable" constraint.

5 Examples

In the following subsections we describe how the general algorithm presented above can be used in two exploratory tasks: 1. investigating the relationship between cardiac muscle fibers and potential distributions, and 2. exploring connectivity of brain white matter in diffusion tensor MRI data.

5.1 Tracing Heart Muscle Fibers

Streamline advection, *i.e.* integrating the motion of massless particles with velocities defined by the field, is a basic building block of vector and tensor field visualization techniques. The haptic equivalent is achieved by constraining proxy motion along the path of a single particle (Figure 1(b)).

This method is easily modified to display orientation information on isosurfaces. Such a technique could be useful for investigating the relationship between heart muscle fiber orientations and potential distributions resulting from cardiac bioelectric finite element simulations [Nielsen et al. 1991]. These simulations are typically carried out on a curvilinear grid that forms a number of epicardial and endocardial layers.

In our implementation we adapt recently developed point location techniques for haptic exploration of data on unstructured tetrahedral grids [Novoselov et al. 2002]. First we compute a Delaunay triangulation of the original datapoints and assign a scalar value to the nodes in individual layers in increasing order from inside to outside. Isosurfaces of this scalar field correspond to muscle layers in the model. Next we find the gradient field using a central difference approximation formula. This field is used in the iterative refinement to make sure the proxy stays on the currently selected layer. Finally, to guarantee smooth interpolation of the fiber orientation vectors and to resolve directional ambiguities, we use componentwise linear interpolation of the tensor field obtained by taking the outer product of the vectors with themselves. The major eigenvector of the interpolated tensor yields a smooth orientation field within a tetrahedral element, even when the vectors at the nodes point in completely different directions.

In this example, a local reference frame is formed by the interpolated fiber orientation and gradient vectors. By applying the snap-drag motion rule, we allow the user to explore a single layer and "pop through" to a neighboring layer by pushing against the surface. In this case the drag threshold τ_i is not used for moving the proxy after breaking away from the current surface. Instead, we detect when the probe crosses a neighboring layer and set the proxy location to a numerical approximation of the intersection point. A secondary snap-drag constraint is used to render fibers on the surface such that the user is able to switch to a nearby streamline in discrete steps. This in turn feels as if the surface was textured with tiny valleys and ridges that correspond to the muscle fibers. See the figure below for an illustration of this example.

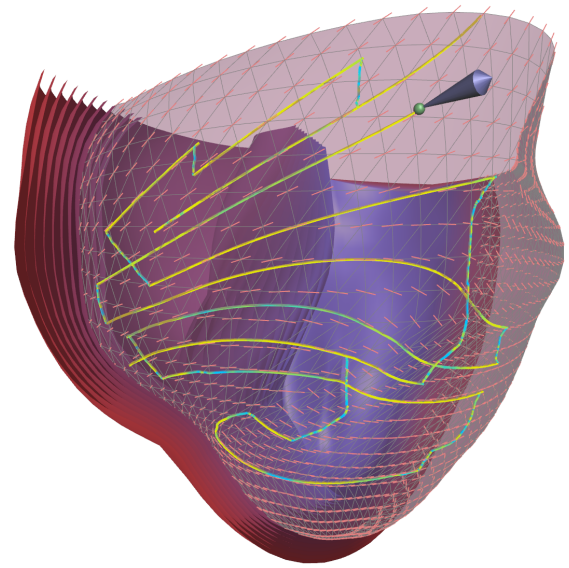


Figure 5: Exploring epicardial muscle fibers with haptic feedback. The probe is constrained to follow local fiber orientation on the surface of a single layer. The user can "pop through" to a neighboring layer by pushing against the surface. Similarly, the user can choose a different fiber by pushing perpendicular to the current one while staying on the surface. This effect feels as if the surface was textured with tiny valleys and ridges. The image shows the path of the proxy colored according to the magnitude of the applied force component perpendicular to the fiber orientation and tangent to the surface, from yellow to cyan indicating increasing tension between the probe and the proxy. The dataset consists of about 30,000 nodes and 200,000 tetrahedral elements.

5.2 Exploring Diffusion Tensor Fields

Diffusion tensor fields are difficult to visualize due to the increased dimensionality of the data values and complexity of the features involved. Direct methods, such as glyphs and reaction-diffusion textures work well on two dimensional slices, but are less successful for creating three dimensional visualizations. Intermediate representations created by adaptations of vector field techniques result in intuitive visual representations, but fail to capture every aspect of the field. Interactive exploration has been reported to be an effective means for helping users interpret the complex geometric models that represent features in the data [Zhang et al. 2001]. Our goal is to aid the exploration process by adding haptic feedback that guides the user according to the local orientation and anisotropy of the tensor field.

The rate and directionality of water diffusion in tissues is indicated by a second-order symmetric tensor. Anisotropy of the diffusion process can be characterized by the following barycentric measures [Kindlmann et al. 2000]:

$$c_l = \frac{\lambda_1 - \lambda_2}{\lambda_1 + \lambda_2 + \lambda_3} \quad (13)$$

$$c_p = \frac{2(\lambda_2 - \lambda_3)}{\lambda_1 + \lambda_2 + \lambda_3} \quad (14)$$

$$c_s = \frac{3\lambda_3}{\lambda_1 + \lambda_2 + \lambda_3} = 1 - c_l - c_p \quad (15)$$

where $\lambda_1 \geq \lambda_2 \geq \lambda_3$ are the sorted eigenvalues of the diffusion tensor matrix. These measures indicate the degree of linear, planar, and spherical anisotropy, respectively. The associated eigenvectors $\vec{e}_1$, $\vec{e}_2$, $\vec{e}_3$ form an orthonormal frame corresponding to the directionality of diffusion. Regions with linear and planar anisotropy represent important features in the data, such as white matter fiber bundles in brain tissue.

One way to use haptic feedback to indicate tensor orientation as well as degree of anisotropy is to control the proxy motion such that it is allowed to move freely along the major eigenvector, but is constrained in the other two directions. We found that setting the drag thresholds according to the anisotropy measures results in the desired feedback:

$$\tau_1 = 0 \quad \tau_2 = \tau(c_l) \quad \tau_3 = \tau(c_l + c_p) \quad (16)$$

where $\tau(x)$ is a monotonically increasing function on $[0 \ldots 1]$. The reasoning behind this particular choice is that the transfer functions should yield a line constraint along the major eigenvector in regions with linear anisotropy ($c_l \gg c_p, c_s$), a plane constraint in regions with planar anisotropy ($c_p \gg c_l, c_s$), and allow free motion along all three directions in isotropic areas ($c_s \gg c_p, c_l$). Recall that the three indices sum to one, so when any one index dominates, the transfer functions will emphasize the corresponding type of anisotropy. Alternatively, we can set the threshold to a constant value for all three directions and vary the stiffness similarly to (16). In our implementation we chose a simple linear ramp for $\tau(x)$, but other possibilities may be more appropriate.

The technique is illustrated in Figure 6. We have observed that it takes relatively little effort to trace out curves indicating fiber distribution and connectivity. Note that numerical methods for fiber tractography require careful selection of initial and stopping conditions and are not straightforward to use for investigating connectivity of regions in the data.

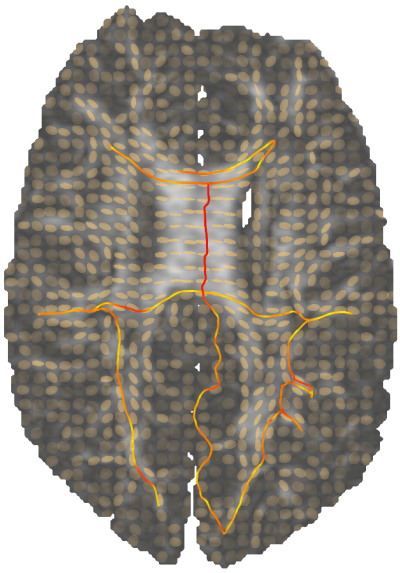


Figure 6: Exploring a 148×190 DT-MRI slice with haptic feedback. The ellipses represent local diffusion anisotropy and orientation. Lighter areas have higher associated anisotropy. The proxy path is colored according to the magnitude of the applied force, from yellow to red indicating a larger tension between the probe and the proxy. The curves are tangent to the direction of the major eigenvector of the diffusion tensor matrix in anisotropic areas.

6 Summary and Future Work

We have described a general constraint-based algorithm for haptic rendering of volumetric datasets. Our approach has several desired properties: it provides a unified rendering framework for different data modalities, secondary effects such as texture and friction are easily realized, haptic transfer functions are intrinsic to the algorithm, and control parameters can be tuned to the operational characteristics of the interface device.

A particular challenge we intend to address in the future is the issue of synthesizing useful haptic transfer functions from the underlying data. Investigating the synergistic relationship between visual and haptic transfer functions is another interesting research topic. A disadvantage of using the spring-damper form of virtual coupling is that it is too conservative, meaning that it may limit the effective display of subtle haptic effects. We will experiment with a recent energy-based approach that uses a time-domain passivity observer and controller to adaptively adjust the coupling parameters [Hannaford and Ryu 2002].

To include tooltip torque into constrained haptic rendering we plan to expand the method for use with 6DOF devices. Transforming the constraint-based approach to a purely functional formulation for haptic scientific visualization would provide a very natural space in which to specify appropriate rendering parameters. Finally, a real challenge for synergistic data display is validation. We intend to quantify the usability of our methods and identify specific techniques useful to scientists who may benefit directly from synergistic display of their datasets.

Acknowledgments

The first author is grateful to Gordon Kindlmann for enlightening and refreshing discussions as well as the `teem` toolkit (`http://teem.sourceforge.net/`), which allowed efficient dataset probing and manipulation in this work. We thank Robert MacLeod for suggesting the heart visualization example and the anonymous reviewers for their helpful comments. The heart model is courtesy of the Bioengineering Institute at the University of Auckland. The diffusion tensor dataset is courtesy of Gordon Kindlmann and Andrew Alexander, W. M. Keck Laboratory for Functional Brain Imaging and Behavior, University of Wisconsin-Madison. Support for this research was provided by NSF Grant ACI-9978063 and the DOE Advanced Visualization Technology Center.

References

ADAMS, R. J., AND HANNAFORD, B. 1998. A two-port framework for the design of unconditionally stable haptic interfaces. In *Proceedings of IEEE International Conference on Intelligent Robots and Systems*, 1254–1259.

AVILA, R. S., AND SOBIERAJSKI, L. M. 1996. A haptic interaction method for volume visualization. In *Proceedings of IEEE Visualization*, 197–204.

BASDOGAN, C., HO, C.-H., SRINIVASAN, M. A., AND SLATER, M. 1998. The role of haptic communication in shared virtual environments. In *Proceedings of PHANToM Users Group Workshop*.

BLEZEK, D. J., AND ROBB, R. A. 1999. Haptic rendering of isosurfaces directly from medical images. In *Proceedings of Medicine Meets Virtual Reality*, 67–73.

BREDERSON, J. D., IKITS, M., JOHNSON, C. R., AND HANSEN, C. D. 2000. The Visual Haptic Workbench. In *Proceedings of PHANToM Users Group Workshop*.

BROOKS, F. P., OUH-YOUNG, M., BATTER, J. J., AND KILPATRICK, P. J. 1990. Project GROPE – Haptic displays for scientific visualization. In *Proceedings of ACM SIGGRAPH*, 177–185.

COLGATE, J. E., AND BROWN, J. M. 1995. Issues in the haptic display of tool use. In *Proceedings of IEEE International Conference on Intelligent Robots and Systems*, 140–145.

DONALD, B. R., AND HENLE, F. 2000. Using haptic vector fields for animation motion control. In *Proceedings of IEEE International Conference on Robotics and Automation*, 3435–3442.

DURLACH, N. I., AND MAVOR, A. S., Eds. 1994. *Virtual Reality: Scientific and Technological Challenges*. National Academy Press, Washington, D.C.

GUNN, C., AND MARANDO, P. 1999. Experiments on the haptic rendering of constraints: Guiding the user. In *Proceedings of Advanced Simulation Technology and Training Conference*.

HANNAFORD, B., AND RYU, J.-H. 2002. Time domain passivity control of haptic interfaces. *IEEE Transactions on Robotics and Automation 18*, 1, 1–10.

HUTCHINS, M. 2000. A constraint equation algebra as a basis for haptic rendering. In *Proceedings of PHANToM Users Group Workshop*.

INFED, F., BROWN, S. V., LEE, C. D., LAWRENCE, D. A., DOUGHERTY, A. M., AND PAO, L. Y. 1999. Combined visual/haptic rendering modes for scientific visualization. In *Proceedings of ASME Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems*, 93–99.

IWATA, H., AND NOMA, H. 1993. Volume haptization. In *Proceedings of IEEE Virtual Reality Annual International Symposium*, 16–23.

KIM, Y. J., OTADUY, M. A., LIN, M. C., AND MANOCHA, D. 2002. Six-degree-of freedom haptic display using localized contact computations. In *Proceedings of IEEE Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems*, 209–216.

KINDLMANN, G. L., WEINSTEIN, D. M., AND HART, D. A. 2000. Strategies for direct volume rendering of diffusion tensor fields. *IEEE Transactions on Visualization and Computer Graphics 6*, 2, 124–138.

KŘENEK, A. 2000. Haptic rendering of molecular flexibility. In *Proceedings of PHANToM Users Research Symposium*.

LAWRENCE, D. A., LEE, C. D., PAO, L. Y., AND NOVOSELOV, R. Y. 2000. Shock and vortex visualization using a combined visual/haptic interface. In *Proceedings of IEEE Visualization*, 131–137.

LUNDIN, K. 2001. *Natural Haptic Feedback from Volumetric Density Data*. Master's thesis, Linköping University, Sweden.

MASCARENHAS, A., EHMANN, S., GREGORY, A., LIN, M., AND MANOCHA, D. 2002. *Touch In Virtual Environments: Haptics and the Design of Interactive Systems*. Prentice-Hall, ch. 5: Six degree-of-freedom haptic visualization, 95–118.

MASSIE, T. H. 1993. *Design of a Three Degree of Freedom Force-Reflecting Haptic Interface*. Bachelor's thesis, Massachusetts Institute of Technology.

MASSIE, T. H. 1996. *Initial Haptic Explorations with the PHANToM: Virtual Touch Through Point Interaction*. Master's thesis, Massachusetts Institute of Technology.

MCNEELY, W. A., PUTERBAUGH, K. D., AND TROY, J. J. 1999. Six degree-of-freedom haptic rendering using voxel sampling. In *Proceedings of ACM SIGGRAPH*, 401–408.

MILLER, T., AND ZELEZNIK, R. C. 1998. An insidious haptic invasion: Adding force feedback to the X desktop. In *Proceedings of ACM User Interface Software and Technology*, 59–64.

NIELSEN, P. M. F., LEGRICE, I. J., SMAILL, B. H., AND HUNTER, P. J. 1991. Mathematical model of geometry and fibrous structure of the heart. In *American Journal of Physiology*, vol. 260, H1365–H1378.

NOVOSELOV, R. Y., LAWRENCE, D. A., AND PAO, L. Y. 2002. Haptic rendering of data on unstructured tetrahedral grids. In *Proceedings of IEEE Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems*, 193–200.

OKAMURA, A. M., DENNERLEIN, J. T., AND HOWE, R. D. 1998. Vibration feedback models for virtual environments. In *Proceedings of IEEE International Conference on Robotics and Automation*, 2485–2490.

OTADUY, M. A., AND LIN, M. C. 2003. Sensation preserving simplification for haptic rendering. *ACM Transactions on Graphics 22*, 3, 543–553.

PAO, L. Y., AND LAWRENCE, D. A. 1998. Synergistic visual/haptic computer interfaces. In *Proceedings of Japan/USA/Vietnam Workshop on Research and Education in Systems, Computation, and Control Engineering*, 155–162.

PETERSIK, A., PFLESSER, B., TIEDE, U., HÖHNE, K. H., AND LEUWER, R. 2002. Haptic volume interaction with anatomic models at sub-voxel resolution. In *Proceedings of IEEE Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems*, 66–72.

RENZ, M., PREUSCHE, C., PÖTKE, M., KRIEGEL, H.-P., AND HIRZINGER, G. 2001. Stable haptic interaction with virtual environments using an adapted voxmap-pointshell algorithm. In *Proceedings of Eurohaptics*.

ROSENBERG, L. B. 1993. Virtual fixtures: Perceptual tools for telerobotic manipulation. In *Proceedings of IEEE Virtual Reality Annual International Symposium*, 76–82.

RUSPINI, D. C., KOLAROV, K., AND KHATIB, O. 1997. The haptic display of complex graphical environments. In *Proceedings of ACM SIGGRAPH*, 345–352.

SALISBURY, J. K., AND TARR, C. 1997. Haptic rendering of surfaces defined by implicit functions. In *Proceedings of ASME Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems*, 61–67.

SALISBURY, K., BROCK, D., MASSIE, T., SWARUP, N., AND ZILLES, C. 1995. Haptic rendering: Programming touch interaction with virtual objects. In *Proceedings of ACM Symposium on Interactive 3D Graphics*, 123–130.

SANKARANARAYANAN, G., DEVARAJAN, V., EBERHART, R., AND JONES, D. B. 2002. Adaptive hybrid interpolation techniques for direct haptic rendering of isosurfaces. In *Proceedings of Medicine Meets Virtual Reality*, 448–454.

SNIBBE, S., ANDERSON, S., AND VERPLANK, B. 1998. Springs and constraints for 3D drawing. In *Proceedings of PHANToM Users Group Workshop*.

SNIBBE, S. S., MACLEAN, K. E., SHAW, R., RODERICK, J., VERPLANK, W. L., AND SCHEEFF, M. 2001. Haptic techniques for media control. In *Proceedings of ACM User Interface Software and Technology*, 199–208.

SRINIVASAN, M. A., AND BASDOGAN, C. 1997. Haptics in virtual environments: Taxonomy, research status, and challenges. *Computer and Graphics 21*, 4, 393–404.

VAN REIMERSDAHL, T., BLEY, F., KUHLEN, T., AND BISCHOF, C. 2003. Haptic rendering techniques for the interactive exploration of CFD datasets in virtual environments. In *Proceedings of Eurographics Workshop on Virtual Environments*, 241–246.

WALL, S. A., AND HARWIN, W. S. 2000. Quantification of the effects of haptic feedback during a motor skills task in a simulated environment. In *Proceedings of PHANToM Users Research Symposium*.

ZHANG, S., DEMIRALP, C., KEEFE, D., DASILVA, M., LAIDLAW, D. H., GREENBERG, B. D., BASSER, P., PIERPAOLI, C., CHIOCCA, E., AND DEISBOECK, T. 2001. An immersive virtual environment for DT-MRI volume visualization applications: A case study. In *Proceedings of IEEE Visualization*, 437–440.

ZILLES, C. B., AND SALISBURY, J. K. 1995. A constraint-based god-object method for haptic display. In *Proceedings of IEEE International Conference on Intelligent Robots and Systems*, 146–151.