

Anisotropic Ambient Volume Shading

Marco Ament, *Student Member, IEEE* and Carsten Dachsbacher

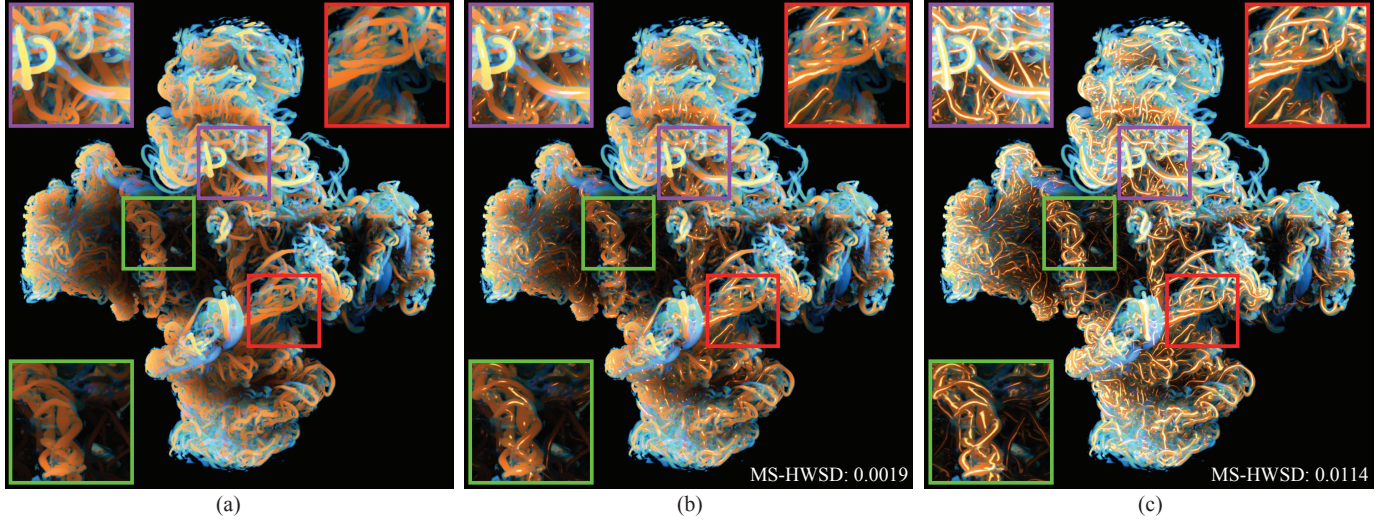


Fig. 1. Volume visualizations of the Vortex data set showing the scalar λ_2 vortex criterion with different reflectance models for illumination. (a) Volumetric scattering with the Henyey-Greenstein phase function without any glossy components. (b) Additional specular highlights using Blinn-Phong's isotropic model. (c) Our approach with anisotropic specular highlights using Laforune's reflectance model, where highlights are automatically aligned with the principal direction of anisotropy of the tubular structures.

Abstract—We present a novel method to compute anisotropic shading for direct volume rendering to improve the perception of the orientation and shape of surface-like structures. We determine the scale-aware anisotropy of a shading point by analyzing its ambient region. We sample adjacent points with similar scalar values to perform a principal component analysis by computing the eigenvectors and eigenvalues of the covariance matrix. In particular, we estimate the tangent directions, which serve as the tangent frame for anisotropic bidirectional reflectance distribution functions. Moreover, we exploit the ratio of the eigenvalues to measure the magnitude of the anisotropy at each shading point. Altogether, this allows us to model a data-driven, smooth transition from isotropic to strongly anisotropic volume shading. In this way, the shape of volumetric features can be enhanced significantly by aligning specular highlights along the principal direction of anisotropy. Our algorithm is independent of the transfer function, which allows us to compute all shading parameters once and store them with the data set. We integrated our method in a GPU-based volume renderer, which offers interactive control of the transfer function, light source positions, and viewpoint. Our results demonstrate the benefit of anisotropic shading for visualization to achieve data-driven local illumination for improved perception compared to isotropic shading.

Index Terms—Direct volume rendering, volume illumination, anisotropic shading

1 INTRODUCTION

Volumetric shading with local illumination models can provide important visual cues to improve the perception of orientation, curvature, and shape of surface-like structures in volumetric data sets. This is motivated by the human visual system, which is capable to reliably estimate the properties of materials by pure visual observations [1]. In particular, specular reflections help convey the shape of surfaces [7, 13]. Probably the most commonly employed local illumination models in direct volume rendering (DVR) are Phong [39] and Blinn-Phong [6] illumination, which exhibit structural information of surfaces [47]. However, only point-wise information is taken into account, which limits the perception of shape on a larger scale.

Alternative ways to convey the shape of surface-like features are the use of other visual representatives than specular highlights, for example, lines [15] or moving particles [31]. Interrante [20] derived a vector field from the principal curvatures of an isosurface to advect virtual particles in tangent space and to compute a stroke texture with a 3D line integral convolution (LIC). However, the dense nature of these techniques can lead to problems in the perceptual segregation of multiple semi-transparent isosurfaces with overlapping patterns, which can be the source of masking effects [12]. Therefore, we opt for a more sparse visual representation based on local illumination.

Kindlmann et al. [24] employed curvature information of isosurfaces to model a multidimensional transfer function. In this approach, the principal curvatures are exploited to control contour thickness for non-photorealistic volume rendering or to emphasize surface variations by directly mapping the principal curvatures to color. However, in this way, color cannot be employed to visually differentiate scalar values, which is an essential element of data classification in volume rendering. In contrast, we employ illumination with specular highlights to visually enhance surface-like structures. A related principle to our approach is the illumination with rim lights to visually enhance the silhouette of an object with a glowing effect, but with our approach, we are not restricted to the silhouette.

• Marco Ament is with Karlsruhe Institute of Technology, Germany.
E-mail: ament@kit.edu.

• Carsten Dachsbacher is with Karlsruhe Institute of Technology, Germany.
E-mail: dachsbacher@kit.edu.

Manuscript received 31 Mar. 2015; accepted 1 Aug. 2015; date of publication xx Aug. 2015; date of current version 25 Oct. 2015.

For information on obtaining reprints of this article, please send e-mail to: tvcg@computer.org.

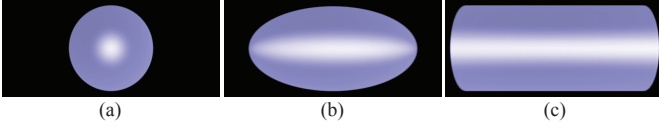


Fig. 2. Volume visualizations of a (a) sphere, (b) ellipsoid, and (c) cylinder. With our contribution, the (an)isotropic shape of the primitives is mapped to (an)isotropic specular highlights for improved perception.

In tensor field visualization [23], the anisotropy of real-valued symmetric tensors is visualized by means of a local illumination model that utilizes the tangent vectors instead of the normal to interpolate between planar and linear anisotropy. In this way, the specular highlights are oriented according to the eigenvectors of the tensor. Moreover, in regions of complete linear anisotropy, shading is identical to the approach by Zöckler et al. [52] for illuminated streamlines. In point-based rendering, Sanftmann and Weiskopf [42] employed the covariance matrix to estimate normals and to visualize different types of anisotropy of three-dimensional scatter plots with color coding. However, in contrast to all these methods, we focus on scalar fields.

Our contribution is a novel method to automatically estimate shading parameters for anisotropic bidirectional reflectance distribution functions (BRDFs) from a scalar field as shown in Figure 2. For the isotropic sphere, the highlight remains isotropic as well, but for the ellipsoid and the cylinder, the highlights become anisotropic and support the perception of shape. We make it possible to utilize anisotropic BRDFs for DVR and to determine meaningful parameters from the data set. Among the large body of physically and phenomenologically motivated BRDFs, we choose Ward’s [50] and Lafortune’s [27] reflectance models as representatives of both types. We introduce an algorithm that allows us to estimate scale-dependent anisotropy from features in the ambient region of a shading point. In this way, we map the anisotropic shape of surface-like structures directly to the anisotropy of a local illumination model, which allows us to visually enhance the orientation and shape of complex features compared to isotropic BRDFs. Furthermore, an important property of our algorithm is that anisotropy estimation is independent of the transfer function, which provides interactive data exploration and frame rates.

2 RELATED WORK

Physically-based light transport in participating media is described by the radiative transfer equation (RTE) [8]. However, for volume visualization, simplified optical models [34] are often preferable because they can be solved with higher performance and their parameters often provide improved control over the visual properties. The survey by Jönsson et al. [22] summarizes illumination techniques that are often employed in interactive DVR. However, no previous work is discussed that employs anisotropic specular highlights.

In interactive DVR, gradient-based shading [29] still remains one of the most widely used illumination techniques due to its high performance and simple implementation. Hossain et al. [19] show how high-quality gradients can be estimated from scalar field data, which are then employed as surface normals to compute isotropic glossy highlights. Tao et al. [47] presented a technique to automatically optimize light source positions and parameters, based on a metric that quantitatively measures how well a lighting setup visually conveys structural information. The metric was evaluated positively in a user study and therefore we employ this metric to study the benefit of our approach.

The literature in computer graphics provides several anisotropic BRDF models [27, 35, 50, 40], but their application in volume rendering was not considered so far. Although Kindlmann et al. [23] employed anisotropic highlights and DVR, the authors visualized second-order tensor fields. Bista et al. [5] visualized fourth-order tensor fields with anisotropic spherical harmonics lighting. However, with both latter approaches, anisotropy information was derived from the tensor data, which is not available in our case. Spherical harmonics were also employed by Schussmann and Ma [44] to illuminate and visualize line data. Moreover, several approaches [32, 52] employ the tangent space

of lines to compute specular highlights. However, all these techniques cannot be used for visualizing scalar fields.

In addition to local illumination, it was shown by several user studies [28, 30] that shadows and advanced lighting effects improve the perception of spatial depth and size. Volumetric ambient occlusion [18] computes attenuation in the direct neighborhood of each voxel to obtain local shadow information, which was improved by Schlegel et al. [43] by using summed area tables [9]. Furthermore, Ament et al. [3] also account for scattering effects in the ambient region of a sample point. With our method, we also gather information from the neighborhood of a voxel; however, in contrast to all these methods, we estimate the parameters for an anisotropic BRDF.

Directional shadows provide additional cues for occlusion on a large scale, for example, by means of a shadow volume [4], deep shadow maps [16] or spherical harmonics [26]. In addition, soft shadows can help avoid disturbing illumination patterns. In interactive DVR, many methods build on the single scattering model [34] together with low-pass filtering operations [2, 37, 41, 43, 48]. Alternatively, soft shadows can be also achieved by simulating multiple scattering, for example, with a convection–diffusion model [51] or Monte-Carlo integration [21, 25]. However, while volumetric shadows improve the perception of spatial depth, we focus on the perception of local structures with anisotropic shading.

3 BASICS OF VOLUMETRIC ILLUMINATION

In this section, we briefly recapitulate the notation and basic principles of volumetric illumination and shading. We build our approach on top of the single scattering illumination model [34], which supports directional shadows. However, our technique does not depend on shadows and only addresses local reflection of incoming light.

3.1 Single Scattering

The total radiance $L(x, \omega)$ at position x in direction ω is the sum of the attenuated radiance $L_b(x_b, \omega)$ from a boundary condition at position x_b and the integrated in-scattered radiance $L_i(x, \omega)$:

$$L(x, \omega) = T(x_b, x)L_b(x_b, \omega) + \int_{x_b}^x T(x', x)\sigma_s(x')L_i(x', \omega)dx', \quad (1)$$

where $\sigma_s(x)$ is the scattering coefficient. The transmittance $T(x_1, x_2)$ between any two points x_1 and x_2 is:

$$T(x_1, x_2) = e^{-\int_{x_1}^{x_2} \sigma_t(x')dx'}, \quad (2)$$

where $\sigma_t(x)$ is the extinction coefficient. Furthermore, in Eqn. (1), $L_i(x, \omega)$ describes the amount of radiance that arrives from all directions at point x and that is scattered into direction ω :

$$L_i(x, \omega) = \int_{\Omega} f(x, \omega', \omega)T(x_b, x)L_b(x_b, \omega')d\omega', \quad (3)$$

where Ω denotes the sphere of all directions and the scattering function $f(x, \omega', \omega)$ describes how much radiance is scattered from incoming direction ω' to outgoing direction ω . In contrast to physically based rendering, we employ a phenomenologically motivated scattering function, which is often preferable for visualization [22]:

$$f(x, \omega', \omega) = k_d(x)f_d(x, \omega', \omega) + k_s(x)f_s(x, \omega', \omega)(\omega' \cdot N_{\sigma_t}(x))_+, \quad (4)$$

where $f_d(x, \omega', \omega)$ simulates volumetric scattering with the Henyey–Greenstein [17] phase function, which is weighted with the diffuse coefficient $k_d(x)$. In addition, we explicitly model surface reflection, and in particular glossy reflection, using a BRDF $f_s(x, \omega', \omega)$, which is multiplied with the specular coefficient $k_s(x)$ and with the cosine of the incident direction ω' and the normal vector $N_{\sigma_t}(x)$, restricted to the positive hemisphere as indicated by the subscript $+$ symbol. For rendering, we compute the normal vector from the estimated gradient of the extinction coefficient:

$$N_{\sigma_t}(x) = -\frac{\nabla \sigma_t(x)}{\|\nabla \sigma_t(x)\|}, \quad (5)$$

where σ_t depends on the transfer function. In addition, we restrict specular highlights to surface-like structures by setting the specular coefficient $k_s(x)$ equal to the gradient magnitude of the extinction coefficient:

$$k_s(x) = \|\nabla\sigma_t(x)\|. \quad (6)$$

Furthermore, we set the diffuse coefficient $k_d(x) = 1$. Note that our scattering function $f(x, \omega', \omega)$ is not physically correct, for example, with regard to conservation of energy, but is tailored to reduce the number of parameters for visualization.

3.2 The BRDF

For visualizing different specular highlights, we employ the Blinn–Phong [6], Ward [50], and Lafortune [27] BRDFs that we review briefly for self-consistency. Subsequently, we require the normalized half-vector, which is defined as follows:

$$H(\omega', \omega) = \frac{\omega + \omega'}{\|\omega + \omega'\|}. \quad (7)$$

Furthermore, for the anisotropic Ward and Lafortune BRDFs, we require the normalized tangent vectors $U(x)$ and $V(x)$ that specify the principal directions of anisotropy. Note that they depend on the position x and it is our main contribution to introduce a novel method to derive meaningful directions from the data set. However, in this section, we assume that they are given.

3.2.1 Blinn–Phong

For comparison, we use the isotropic specular component of the Blinn–Phong BRDF for volumetric illumination with:

$$f_{s,b}(x, \omega', \omega) = (N_{\sigma_t}(x) \cdot H(\omega, \omega'))^{\kappa(x)}, \quad (8)$$

where $\kappa(x)$ is the Blinn–Phong exponent that controls the glossiness of the specular highlight.

3.2.2 Ward

Our first anisotropic model is the Ward BRDF:

$$f_{s,w}(x, \omega', \omega) = A_1(x, \omega', \omega) \exp\left(-\frac{A_2(x, \omega', \omega)}{(H(\omega', \omega) \cdot N_{\sigma_t}(x))^2}\right). \quad (9)$$

The terms controlling anisotropy in Eqn. (9) are described by:

$$A_1(x, \omega', \omega) = \frac{1}{4\pi u_w(x) v_w(x) \sqrt{(N_{\sigma_t}(x) \cdot \omega') (N_{\sigma_t}(x) \cdot \omega)}}, \quad (10)$$

$$A_2(x, \omega', \omega) = \left(\frac{H(\omega', \omega) \cdot U(x)}{u_w(x)}\right)^2 + \left(\frac{H(\omega', \omega) \cdot V(x)}{v_w(x)}\right)^2. \quad (11)$$

In addition to the tangent vectors, the Ward BRDF requires specific parameters $u_w(x)$ and $v_w(x)$ that control the magnitudes of the anisotropy in both tangent directions, respectively. They both depend on the position x and one of our contributions is to derive meaningful values from the data set.

3.2.3 Lafortune

From the class of phenomenological BRDFs, we use the specular component of Lafortune’s BRDF:

$$f_{s,l}(x, \omega', \omega) = (u_l(x)\tilde{\omega}'_u\tilde{\omega}_u + v_l(x)\tilde{\omega}'_v\tilde{\omega}_v + n_l(x)\tilde{\omega}'_n\tilde{\omega}_n)^{\kappa_l(x)}, \quad (12)$$

where $\kappa_l(x)$ is similar to the Blinn–Phong exponent, but we employ an additional subscript l to highlight its association to Lafortune’s BRDF. Moreover, $\tilde{\omega}' = (\tilde{\omega}'_u, \tilde{\omega}'_v, \tilde{\omega}'_n)^T$ and $\tilde{\omega} = (\tilde{\omega}_u, \tilde{\omega}_v, \tilde{\omega}_n)^T$ denote the incoming and outgoing directions with respect to the local orthonormal frame $U(x)$, $V(x)$, and $N_{\sigma_t}(x)$. In addition, the Lafortune BRDF provides specific parameters $u_l(x)$, $v_l(x)$, and $n_l(x)$ to control the shape of the specular highlight. For a detailed discussion on these parameters, the reader is referred to the thesis by McAllister [35]. For the remainder of this paper, we set $n_l(x) = 1$, and one of our contributions is to compute meaningful values for $u_l(x)$ and $v_l(x)$ from the data set.

4 THE TECHNIQUE

In this section, we present our novel technique for anisotropic volume shading. In Section 4.1, we show how a local orthonormal frame of reference is estimated from the data set, in particular, the tangent vectors $U(x)$ and $V(x)$ for the principal anisotropy directions. Moreover, in Section 4.2, we show how anisotropy is mapped to the Ward-specific parameters $u_w(x)$ and $v_w(x)$ as well as the Lafortune-specific parameters $u_l(x)$ and $v_l(x)$. Finally, in Section 4.3, we provide a discussion of our approach with regard to parameters and our design choices.

4.1 Anisotropy Direction

We assume that volume data is stored in the form of a discrete Cartesian lattice that samples a scalar field $s: \mathbb{R}^3 \rightarrow \mathbb{R}$. On a high level of abstraction, our algorithm first computes high-quality normals from the unclassified scalar field s , which we briefly discuss in Section 4.1.1. These normals then define local tangent planes at each point x and we restrict the further computation of tangent vectors $U(x)$ and $V(x)$ to this plane, which we explain in detail in Section 4.1.2. In this way, our algorithm is independent of the transfer function and we can exploit previous research in gradient estimation [10, 19, 36, 45] to obtain high-quality normals, which is crucial for artifact-free shading.

4.1.1 Normal Vector Estimation

In contrast to the normal estimation in Eqn. (5) for rendering, we now estimate normals from the gradient of the unclassified scalar field s :

$$N(x) = -\frac{\nabla s(x)}{\|\nabla s(x)\|}, \quad (13)$$

where $\nabla s(x)$ can be any high-quality gradient estimator; subsequently, we employ the Sobel operator [45]. Using $N(x)$ instead of $N_{\sigma_t}(x)$ has the advantage that we can compute and filter gradients once and store them on a grid, independent of the transfer function. It is important to note that in many cases $N(x)$ and $N_{\sigma_t}(x)$ can differ only in their signs; however, this does not affect the tangent plane. Furthermore, with a locally constant scalar field, we obtain $\|\nabla s(x)\| = 0$ and we skip normal and tangent estimation. In this case, it also follows that $\|\nabla\sigma_t(x)\| = 0$, independent of the transfer function and the specular coefficient becomes $k_s = 0$, according to Eqn. (6). The latter observation also follows for a locally constant transfer function. Therefore, in both cases, specular shading is not applied according to Eqn. (4).

4.1.2 Tangent Vector Estimation

After normal estimation, we restrict the computation of both tangent vectors to tangent space. We build our approach of estimating tangent vectors on a covariance analysis of the ambient neighborhood of a point x . Similarly, several previous papers on point-based rendering estimate principal curvatures or normal vectors of a set of points, for example, for mesh simplification [14, 38] or anisotropy classification [42]. However, in contrast, we demonstrate how a covariance analysis can be adapted to volumetric data and how its result can be exploited to estimate high-quality tangent vectors for an anisotropic BRDF.

Ambience Exploration The first step of our algorithm in computing tangent vectors at a point x is an explorative sampling of its ambient region. The goal of this step is to obtain a set of points $X_a(x)$ that are uniformly distributed in the interval volume $[s(x) - \varepsilon, s(x) + \varepsilon]$, where $s(x)$ is the scalar value at point x and $\varepsilon > 0$ is a small number.

Therefore, we define a sphere of radius r that is centered at x and we draw n samples of a probability density function that generates a set of 3D positions $X(x) = \bigcup_{i=1}^n x_i$ of uniform density inside this sphere as illustrated in Figure 3(a). To obtain $x_i = (x_{i,x}, x_{i,y}, x_{i,z})^T$, we first draw a random number $\xi_0 \in [0, 1]$ from a uniform distribution to compute the distance $d \leq r$ from the center of the sphere:

$$d = r\sqrt[3]{\xi_0}. \quad (14)$$

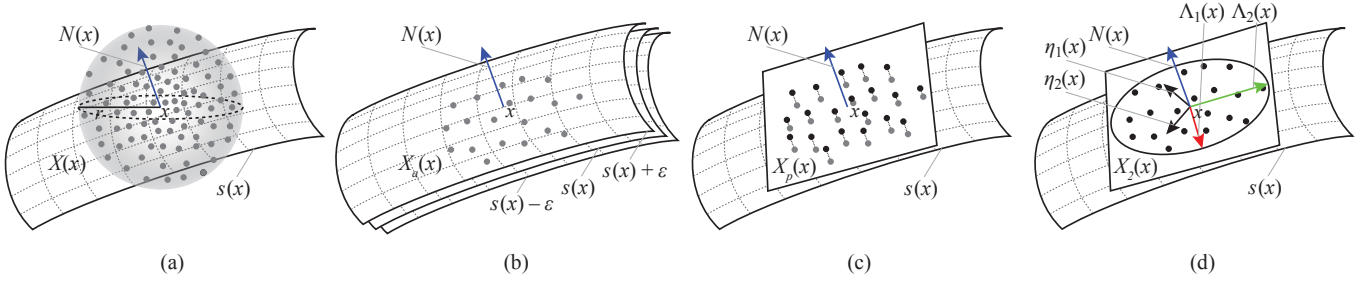


Fig. 3. Illustration of tangent vector estimation at point x with normal vector $N(x)$. The scalar value $s(x)$ defines an isosurface, which we want to sample. (a) Computation of the point set $X(x)$ by explorative sampling of the ambient region of x inside a sphere of radius r . (b) Computation of the point set $X_a(x)$ by taking all samples of $X(x)$ that lie inside the interval volume defined by $[s(x) - \varepsilon, s(x) + \varepsilon]$. (c) Computation of the point set $X_p(x)$ by projecting all samples of $X_a(x)$ onto the tangent plane defined by the normal vector. (d) Computation of the point set $X_2(x)$ by transforming all samples of $X_p(x)$ into 2D space with $\eta_1(x)$ and $\eta_2(x)$ as local frame of reference. Then, the 2×2 covariance matrix is obtained from $X_2(x)$ and the eigenvectors $\Lambda_1(x)$ and $\Lambda_2(x)$ describe the minor and major directions of anisotropy, respectively.

Second, we draw two additional random numbers $\xi_1, \xi_2 \in [0, 1]$ from a uniform distribution to compute the position on the surface of the sphere with center $x = (x_x, x_y, x_z)^T$ and radius d [11]:

$$x_{i,x} = x_x + 2d \cos(2\pi\xi_1) \sqrt{\xi_2(1-\xi_2)} \quad (15)$$

$$x_{i,y} = x_y + 2d \sin(2\pi\xi_1) \sqrt{\xi_2(1-\xi_2)} \quad (16)$$

$$x_{i,z} = x_z + d(1-2\xi_2). \quad (17)$$

With these samples, we compute the set of positions $X_a(x)$ that have a similar scalar value as $s(x)$ inside the sphere as shown in Figure 3(b):

$$X_a(x) = \{x_i : x_i \in X(x), s(x_i) \in [s(x) - \varepsilon, s(x) + \varepsilon]\}. \quad (18)$$

$X_a(x)$ is a subset of $X(x)$ with $|X_a(x)| = m \leq n$ and its elements are obtained by testing if the scalar value $s(x_i)$ of all elements $x_i \in X(x)$ lies within the finite interval $[s(x) - \varepsilon, s(x) + \varepsilon]$; we call $X_a(x)$ the set of accepted samples. In the limits $\varepsilon \rightarrow 0$ and $n \rightarrow \infty$, $X_a(x)$ contains the set of all points that describe an isosurface with isovalue $s(x)$ inside the ambient sphere.

Projection The second step of our algorithm reduces the problem of tangent vector estimation to the tangent space, defined by the previously computed high-quality normals. Therefore, the set of accepted samples $X_a(x)$ is projected onto the tangent plane as shown in Figure 3(c):

$$X_p(x) = \{x_i : x_i = x - (N(x) \cdot (y_i - x))N(x), y_i \in X_a(x)\}. \quad (19)$$

Subsequently, the projected 3D points are transformed into a local 2D coordinate system to reduce anisotropy estimation to two dimensions as illustrated in Figure 3(d). For this reason, we require a deterministic and cheap method to compute a local frame of reference. In particular, we need two arbitrary orthonormal vectors $\eta_1(x)$ and $\eta_2(x)$ in the tangent plane. Furthermore, we seek to avoid discontinuities of $\eta_1(x)$ and $\eta_2(x)$ between adjacent samples, which could be a source of artifacts due to filtering. Therefore, we interpret the normal vector $N(x)$ as a position $(\phi(x), \theta(x))$ on the unit sphere. In this case, the radial unit vector of this position is equal to the normal vector $N(x)$. Consequently, we obtain an orthonormal vector $\eta_1(x)$ with $N(x) \cdot \eta_1(x) = 0$ by computing the unit vector $e_\phi(x)$ in azimuthal direction. Then, the second orthonormal vector $\eta_2(x)$ is computed by a cross product:

$$\eta_1(x) = e_\phi(x) = -\sin(\phi(N(x)))e_x + \cos(\phi(N(x)))e_y \quad (20)$$

$$\eta_2(x) = \eta_1(x) \times N(x), \quad (21)$$

where $e_x = (1, 0, 0)^T$ and $e_y = (0, 1, 0)^T$ are unit vectors in Cartesian space and $\phi(N(x)) = \text{atan2}(N_y(x), N_x(x))$. Finally, the set of projected samples $X_p(x)$ is transformed to 2D:

$$X_2(x) = \{x_i : x_i = (y_i \cdot \eta_1(x), y_i \cdot \eta_2(x))^T, y_i \in X_p(x)\}. \quad (22)$$

Note that $X_2(x)$ can be obtained directly without an explicit computation of $X_p(x)$.

Covariance Matrix The third step of our algorithm employs the set of projected 2D samples X_2 to build a distance-weighted covariance matrix $C(x)$:

$$C(x) = \sum_{i=1}^m (x_i - \bar{x})(x_i - \bar{x})^T w\left(\frac{\|x_i - \bar{x}\|}{r}\right), \quad x_i \in X_2(x), \quad (23)$$

where \bar{x} is the centroid of the m elements in $X_2(x)$:

$$\bar{x} = \frac{1}{m} \sum_{i=1}^m x_i, \quad x_i \in X_2(x). \quad (24)$$

Moreover, $w(d)$ is a monotonically decreasing weighting function. In this paper, we use the following function:

$$w(d) = \begin{cases} 1 - d^2 & \text{if } d \leq 1 \\ 0 & \text{else.} \end{cases} \quad (25)$$

Note that $C(x)$ is only a 2×2 matrix, because all $x_i \in \mathbb{R}^2$ and moreover the matrix is symmetric and can thus be represented with only 3 values $C_{11}(x)$, $C_{12}(x)$, and $C_{22}(x)$.

Filtering Since the computation of the covariance matrix is restricted to tangent space, which is obtained from high-quality normal vectors, it is often possible to extract eigenvectors from $C(x)$ without additional filtering. However, for small radii r and data sets with a low signal-to-noise ratio, it can be necessary to perform additional filtering of the tangent vectors. In principle, such filtering would be applied after extracting the eigenvectors of $C(x)$; however, filtering the elements $C_{11}(x)$, $C_{12}(x)$, and $C_{22}(x)$ of the covariance matrix instead of the eigenvectors has the advantage that no orthogonalization is required after filtering. Therefore, we introduce an optional filtering pass of the covariance matrix using a discrete Gaussian filter kernel, before computing the eigenvectors.

Eigenanalysis Since $C(x)$ is a positive semi-definite matrix, the eigenvalues $\lambda_1(x)$ and $\lambda_2(x)$ are non-negative and real-valued. Both eigenvalues can be computed from the trace and determinant of the covariance matrix:

$$\lambda_{1,2}(x) = \frac{\text{tr}(C(x))}{2} \pm \sqrt{\frac{\text{tr}(C(x))^2}{4} - \det(C(x))}. \quad (26)$$

In addition, the corresponding eigenvectors $\Lambda_1(x)$ and $\Lambda_2(x)$ form an orthogonal frame in tangent space, describing the principal components of the point set X_2 . If $C_{12}(x) \neq 0$, we get:

$$\Lambda_1(x) = (C_{12}(x), \lambda_1 - C_{11}(x))^T, \quad (27)$$

$$\Lambda_2(x) = (C_{12}(x), \lambda_2 - C_{11}(x))^T. \quad (28)$$

If $C_{12}(x) = 0$, it follows that $\Lambda_1(x) = (1, 0)^T$ and $\Lambda_2(x) = (0, 1)^T$ with respect to $\eta_1(x)$ and $\eta_2(x)$, describing an isotropic distribution. Subsequently, we assume that $\lambda_1(x) \leq \lambda_2(x)$ or otherwise we swap both values. Then, $\Lambda_2(x)$ points in the direction of the strongest variance of the point set $X_2(x)$ which we identify with the major direction of anisotropy and $\Lambda_1(x)$ with the minor direction. However, both eigenvectors are still given with respect to $\eta_1(x)$ and $\eta_2(x)$ and we need to transform them back to 3D world space:

$$\bar{\Lambda}_1(x) = \Lambda_{1,x}(x)\eta_1(x) + \Lambda_{1,y}(x)\eta_2(x) \quad (29)$$

$$\bar{\Lambda}_2(x) = \Lambda_{2,x}(x)\eta_1(x) + \Lambda_{2,y}(x)\eta_2(x). \quad (30)$$

Finally, we obtain the normalized tangent vectors:

$$U(x) = \frac{\bar{\Lambda}_2(x)}{\|\bar{\Lambda}_2(x)\|}, \quad V(x) = \frac{\bar{\Lambda}_1(x)}{\|\bar{\Lambda}_1(x)\|}. \quad (31)$$

Note that $U(x)$ and $V(x)$ can be employed both for Lafortune's and Ward's BRDF; however, the BRDFs have different parameters that control the degree of anisotropy and the glossiness of the specular highlights, which we discuss in the following section.

4.2 Anisotropy Mapping

For both BRDFs, our goal is to model the visual appearance of threads and scratches pointing in the major direction of anisotropy, which we exploit to improve perception of salient structures. In this section, we show how the position-dependent anisotropy parameters $u_w(x), v_w(x)$ and $u_l(x), v_l(x)$ can be obtained from the previous eigenanalysis in conjunction with meaningful user-controlled global parameters. First of all, we define a global parameter κ that controls the glossiness of a specular lobe, just like the Blinn-Phong exponent for isotropic shading. Furthermore, we introduce a global parameter γ for the degree of anisotropy, which controls how strong the stretching of anisotropic highlights is for visualization. Since the anisotropy parameters $u_w(x), v_w(x)$ and $u_l(x), v_l(x)$ are specific to the corresponding BRDF, it is difficult to derive BRDF-independent parameters for κ and γ . Therefore, we also employ BRDF-specific parameters κ_w, γ_w and κ_l, γ_l with their own ranges of values.

4.2.1 Ward

The Ward-specific parameters $u_w(x) > 0, v_w(x) > 0$ in Eqns. (10) and (11) control the glossiness with their magnitude and the degree of anisotropy with their relative ratio. We introduce the following mapping functions to combine the global parameters κ_w, γ_w with the ratio of the eigenvalues, which provides position-dependent information on the degree of anisotropy:

$$u_w(x) = \kappa_w \left(\frac{\lambda_2(x)}{\lambda_1(x)} \right)^{\gamma_w}, \quad (32)$$

$$v_w(x) = \kappa_w \quad (33)$$

with $\kappa_w > 0$ and $\gamma_w \geq 0$. Figure 4(a) shows plots of Eqn. (32) with different scaling factors γ_w . From the eigenanalysis, it follows that $\lambda_1(x) = \lambda_2(x)$ in isotropic regions. In this case, we also get isotropic shading with $u_w(x) = v_w(x) = \kappa_w$ and a user can adjust the glossiness with κ_w . However, in anisotropic regions, we have $\lambda_1(x) < \lambda_2(x)$ and the user-controlled parameter γ_w acts as a scaling factor that amplifies the prevalent anisotropy for $\gamma_w > 1$ and damps it for $\gamma_w < 1$. Moreover, for $\gamma_w = 0$, we obtain again isotropic shading.

4.2.2 Lafortune

With Lafortune's BRDF, the glossiness is directly controlled with the exponent $\kappa_l(x)$, and the anisotropy parameters $u_l(x), v_l(x)$ implement different types of reflection [35]. Again, we employ the ratio of the eigenvalues to exploit position-dependent information on the degree of anisotropy from the data set and we introduce the following mapping

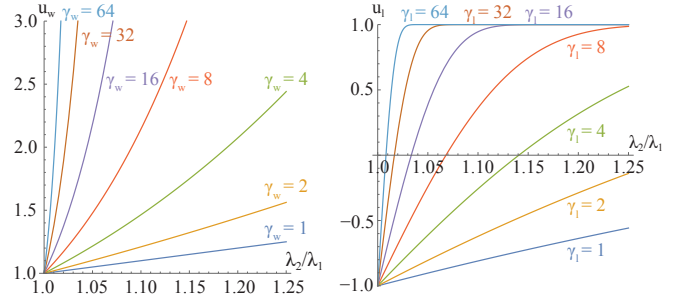


Fig. 4. Plots of the anisotropy mapping functions for (a) Ward's and (b) Lafortune's BRDF with different scaling factors γ_w and γ_l .

functions with the global parameter γ :

$$u_l(x) = 2 \left(1 - \exp \left(1 - \left(\frac{\lambda_2(x)}{\lambda_1(x)} \right)^{\gamma_l} \right) \right) - 1, \quad (34)$$

$$v_l(x) = -1 \quad (35)$$

with $\gamma_l \geq 0$. Figure 4(b) shows plots of Eqn. (34) with different scaling factors γ_l . In isotropic regions of the data set with $\lambda_1(x) = \lambda_2(x)$, we obtain $u_l(x) = v_l(x) = -1$, which models isotropic shading [35] and a user can adjust only the glossiness with κ_l . In anisotropic regions with $\lambda_1(x) < \lambda_2(x)$, the user-controlled parameter γ_l acts again as a scaling factor that amplifies or damps anisotropy for visualization, similar to γ_w for Ward's BRDF. In particular, we also obtain isotropic shading for $\gamma_l = 0$. However, in contrast to Eqn. (32), the useful range of values is $u_l(x), v_l(x) \in [-1, 1]$, but the ratio of the eigenvalues is not bounded. Therefore, we map the possibly infinite range of values to the finite interval $[-1, 1]$ using an exponential function. Together with $v_l(x) = -1$, this setting implements similar reflectance characteristics than Eqns. (32) and (33) for Ward's BRDF.

4.3 Discussion

Before we provide details on our implementation in the next section, we briefly discuss our approach with respect to parameters and design.

4.3.1 Parameters

Subsequently, we focus only on parameters that are specific to our approach of estimating tangent vectors: the radius r of the ambient sphere, the interval size ϵ , and the number of samples n .

The radius r of the ambient sphere controls the feature size in the *spatial domain* for estimating anisotropy. A small radius implements a localized exploration of an isosurface, which provides local curvature information close to the point of interest. However, if the radius becomes too small, additional filtering is necessary to avoid artifacts due to the discrete nature of the data. In contrast, a large radius also accounts for large-scale structures and is less susceptible to discretization artifacts, but if the radius is chosen too large, anisotropy estimation becomes overly isotropic and furthermore the number of samples needs to be increased to maintain a constant sample density.

The interval size ϵ controls the feature size in the *data domain* for estimating anisotropy. A small value for ϵ implements a localized sampling in data space and complements exploration in the spatial domain with a small radius to obtain local anisotropy information for a sharp isosurface. However, too small interval sizes result in noise artifacts, which requires either extensive filtering or a high number of samples to obtain high visual quality. In addition, too few accepted samples can also result in non-uniform clusters, which falsifies anisotropy estimation, because the spatial position of the samples directly influences the covariance matrix and its eigenanalysis. In contrast, large values for ϵ sample a wide interval in data space, which is beneficial for estimating anisotropy of true volumetric structures, not just for isosurfaces. Furthermore, fewer samples are required and filtering can often be omitted. However, choosing too large values for ϵ includes many different features, which results in a loss of anisotropic details.

Algorithm 1 Estimation of normal vectors and covariance matrices.

Require: radius r , interval size ε , number of samples n , Boolean filter flag f

```

1: for all voxels  $x$  do
2:    $N = \text{estimateHighQualityNormal}(x)$ ;
3:    $\text{storeVectorAtVoxel}(N, x)$ ;
4: end for
5: for all voxels  $x$  do
6:    $s = \text{sampleScalarField}(x)$ ;
7:   for  $i = 1$  to  $n$  do
8:      $x_i = \text{drawUniformSampleSphere}(x, r)$ ;
9:      $s_i = \text{sampleScalarField}(x_i)$ ;
10:    if  $s - \varepsilon \leq s_i \leq s + \varepsilon$  then
11:       $x_p = \text{project3DPointToTangentPlane}(x_i, N)$ ;
12:       $x_2 = \text{transform3DPointTo2DPoint}(x_p, N)$ ;
13:       $C = \text{add2DPointToCovarianceMatrix}(x_2, C)$ ;
14:    end if
15:  end for
16:   $\text{storeSymmetric2x2MatrixAtVoxel}(x, C)$ ;
17: end for
18: if  $f == \text{true}$  then
19:   for all voxels  $x$  do
20:      $C = \text{readSymmetric2x2MatrixFromVoxel}(x)$ ;
21:      $C = \text{filterGaussian}(C)$ ;
22:      $\text{storeSymmetric2x2MatrixAtVoxel}(C, x)$ ;
23:   end for
24: end if

```

4.3.2 Design

In principle, tangent vectors could also be estimated from the principal curvatures of an isosurface [24]. However, we decided against a pure local approach for two reasons.

First, estimating principal curvatures requires careful filtering of the scalar field as well as the first and second derivatives with higher-order polynomials [24]. However, even for medium-sized data sets, on-the-fly computation becomes too expensive for interactive visualization. A common alternative is prefiltering, but then the memory footprint becomes problematic, because for computing the geometry tensor, it is necessary to store 3 values per voxel for the gradient and 6 values per voxel for the symmetric 3×3 Hessian matrix. Although our approach also takes 3 values per voxel for the gradient, it only requires 3 values per voxel for the symmetric 2×2 covariance matrix.

The second advantage of our method is that we estimate the anisotropy for structures that have a finite extent in both the spatial and data domain, in contrast to derivative-based curvature computation. While our approach can still estimate local anisotropy for sharp isosurfaces, it is also possible to focus more on true volumetric structures on different scales. In this way, local isotropic areas that are part of a larger anisotropic feature can also be classified as partially anisotropic and shading becomes less susceptible to artifacts due to strong variations of anisotropy or a low signal-to-noise ratio.

5 IMPLEMENTATION

Our CUDA-based implementation on the GPU consists of two major steps. First, we describe all operations that are independent of the transfer function and that we perform only once for each data set. Second, we present our algorithm for DVR, which offers interactive manipulation of the transfer function, view point, and light sources.

5.1 Transfer Function-Independent Computation

The first part of our implementation computes high-quality normals and covariance matrices for each voxel as shown in Algorithm 1. We employ the Sobel operator for gradient estimation and implement the stages of ambience exploration, projection, covariance matrix computation, and optional filtering of Section 4.1.2. We store the prefiltered normals and covariance matrices on two uniform 3D grids, quantizing all values to 8 bits per channel, which usually provides sufficient quality for gradient-based shading [46]. It is important to note that we do not compute and store eigenvectors at this point, because interpolation is difficult and can lead to inconsistent results [23]. Instead, we interpolate the components of the covariance matrix, similar to tensor visualization, and perform the eigenanalysis on-the-fly during rendering as discussed in the next section.

Algorithm 2 DVR with Ward’s BRDF.

Require: transfer function TF , light sources LS , Henyey-Greenstein phase function $f_{d,hg}$, Ward BRDF $f_{s,w}$, background radiance L_b , glossiness κ_w , scaling factor γ_w , diffuse coefficient k_d

```

1: for all pixels  $P$  do
2:    $L = 0$ ;  $T = 0$ ;
3:    $\text{ray} = \text{computeEyeRay}(P)$ ;
4:    $\omega = -\text{ray.direction}$ ;
5:   while  $\text{rayCastingNotFinished}(\text{ray})$  do
6:      $x = \text{getRayPosition}(\text{ray})$ ;
7:      $C = \text{sampleSymmetric2x2MatrixFromPosition}(x)$ ;
8:      $G_{\sigma_i} = \text{sampleGradientFromPosition}(x)$ ;
9:      $N_{\sigma_i} = G_{\sigma_i} / \text{length}(G_{\sigma_i})$ ;
10:     $(\lambda_1, \lambda_2, \Lambda_1, \Lambda_2) = \text{computeEigenanalysis}(C)$ ;
11:     $(U, V) = \text{transformToWorldSpaceAndNormalize}(\Lambda_1, \Lambda_2, N_{\sigma_i})$ ;
12:     $(u_w, v_w) = \text{computeWardParameters}(\lambda_1, \lambda_2, \kappa_w, \gamma_w)$ ;
13:     $k_s = \text{length}(G_{\sigma_i})$ ;
14:     $L_i = 0$ ;
15:    for all  $l_s \in LS$  do
16:       $\omega' = \text{lightDirection}(x, l_s)$ ;
17:       $f = k_d f_{d,hg}(\omega', \omega) + k_s f_{s,w}(U, V, N_{\sigma_i}, u_w, v_w, \omega', \omega)(\omega' \cdot N_{\sigma_i}) +$ 
18:         $L_i = L_i + f \text{ incomingRadianceFromLightSource}(x, l_s)$ ;
19:    end for
20:     $(L, T) = \text{composite}(x, TF, L, T, L_i)$ ;
21:  end while
22:  return  $T \cdot L_b + L$ ;
23: end for

```

5.2 Direct Volume Rendering

For DVR, we perform ray casting from the virtual camera. We employ 3D textures for the data set, prefiltered normals, and covariance matrices. Furthermore, we support optional volumetric shadows by means of a shadow cache. Our rendering method is summarized in Algorithm 2, shown exemplarily for Ward’s BRDF; for LaFortune’s BRDF the algorithm follows analogously. In Algorithm 2, we interpolate the covariance matrix at each ray sample and compute the eigenvalues λ_1, λ_2 and eigenvectors Λ_1, Λ_2 , which are then used to compute the Ward-specific anisotropy parameters u_w, v_w . The specular coefficient k_s is derived from the gradient magnitude and anisotropic shading is performed for each light source. Incoming radiance can be either local or attenuated light for volumetric shadows.

6 RESULTS

We present results with different data sets exhibiting volumetric structures of different scale and shape to evaluate our approach. In particular, we demonstrate the benefit of anisotropic highlights compared to isotropic ones. For a quantitative analysis, we implemented the multi-scale homogeneity-weighted structural dissimilarity (MS-HWSD) metric by Tao et al. [47]. In their work, this metric was exploited to optimize light source positions based on the structural similarity index [49] and information theory. In our work, we show that consistently higher scores can be achieved with anisotropic BRDFs than with isotropic shading for a fixed lighting setup, indicating that anisotropic BRDFs are well-suited to visually convey structural information. All results were obtained with an Intel Core i7 3.4 GHz CPU, 32 GB RAM, and an NVIDIA Titan GPU.

In Figure 5, we first compare our method of estimating anisotropy with the curvature-based approach by Kindlmann et al. [24]. From differential geometry, it is known that the principal curvature directions form an orthonormal basis in the tangent plane. Therefore, we compute the principal curvatures at each shading point and use them as input for Ward’s BRDF. In an effort to provide a comparable implementation in terms of rendering quality, performance, and memory consumption, we precompute and filter the Hessian matrix on a 3D grid, similar to our approach. At rendering, the geometry tensor is computed from the filtered gradient vector and Hessian matrix. The eigenvalues and eigenvectors of the geometry tensor are computed on-the-fly, which correspond to the principal curvature magnitudes and directions, respectively. In Figure 5, we visualize a set of closely packed ellipsoids with decreasing anisotropy from the center toward the boundaries using a data set resolution of $250 \times 92 \times 512$ voxels. In Figure 5(a), anisotropy is estimated from the principal curvatures, which

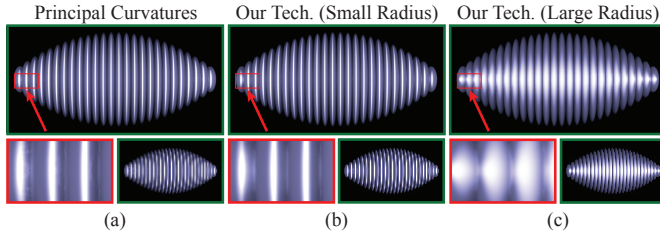


Fig. 5. Visualization of a set of ellipsoids. The anisotropy decreases from the center toward the boundaries and is estimated from (a) principal curvatures [24] and with our technique using a (b) small and (c) large radius. The red inset shows a close-up of one specular highlight. The green inset shows the data set from a far distance (upscaled images).

highlights the vertical shape of the individual ellipsoids. For a direct comparison, we estimate anisotropy with our method in Figure 5(b) by using a small radius of $r = 3$ voxels, which leads to a very similar visualization, but the close-up views reveal that the curvature-based method is susceptible to artifacts close to the transitions of the ellipsoids, in contrast to our method. An advantage of the curvature-based approach is that the precomputation is faster with 465 ms compared to our method with 985 ms to achieve a similar quality. However, memory consumption is higher with 67.4 MB compared to 44.9 MB and rendering is also slower with 69 fps compared to 81 fps with our technique, because a symmetric 3×3 matrix needs to be stored and accessed, in contrast to a symmetric 2×2 matrix in our case.

Common to both visualizations is that anisotropy is estimated locally, which can lead to disturbing visual patterns, for example, when the data set is noisy or when the features are small compared to the image resolution. The green insets in Figures 5(a) and (b) show aliasing patterns when the camera is zoomed-out. A unique aspect of our method is that we can increase the ambient radius to estimate anisotropy on a larger scale to avoid these issues or to highlight the more salient structures of a data set as shown in Figure 5(c) where we increase the radius to $r = 200$ voxels. The specular highlight now visually emphasizes the overall structure of the data set, instead of the individual ellipsoids. Furthermore, when zooming out, aliasing is significantly reduced. This simple example demonstrates that our technique is capable to reliably estimate anisotropy on different scales with comparable resource consumption than the curvature-based approach. In all subsequent results, we employ our method.

In Figure 6, we study the parameters for estimating the covariance matrix of Section 4.1.2: the ambient radius r , the interval size ε , and the number of samples n . For simplicity, we created an artificial data set of an ellipsoid with a resolution of 512^3 voxels. In Figures 6(a) and (b), we show two reference results with large and moderate scaling factors γ_w and γ_l . In both images, the anisotropy of the ellipsoid is visually mapped to the prolated specular highlights and for smaller values of γ_w and γ_l , the stretching is less pronounced. Since the ellipsoid covers almost the entire volume, we use a rather large radius of $r = 200$ voxels and a small interval size of $\varepsilon = 0.01$, assuming $s \in [0, 1]$. In this way, anisotropy estimation is focused on isosurfaces, which requires $n = 500$ samples per voxel for high quality.

In Figures 6(c) and (d), we decreased the ambient radius to $r = 1$ voxel and anisotropy estimation is strongly localized in the spatial domain, similar to curvature computation [24]. The large-scale anisotropy of the ellipsoid can be hardly determined as the surface is almost isotropic on this small scale. However, in Figure 6(c), we still enforce high scaling factors γ_w and γ_l , but this leads to noticeable artifacts. In Figure 6(d), we decrease γ_w and γ_l until the artifacts disappear, but the specular highlights become almost isotropic as opposed to the reference in Figure 6(b).

In Figures 6(e) and (f), we use again the settings of the reference in Figure 6(a), but we increase the interval size to $\varepsilon = 0.5$. In this way, anisotropy estimation is focused on large volumetric features in data space. However, the large range of values now also covers features that are not part of the ellipsoid, which leads to inconsistent anisotropy

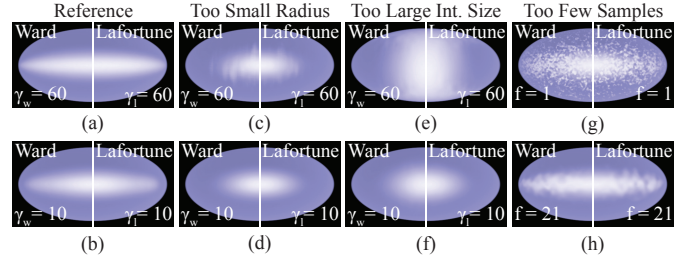


Fig. 6. Visualization of an ellipsoid. Reference visualizations with (a) strong and (b) moderate scaling factors. A too small radius cannot detect the anisotropy and leads to either (c) artifacts with strong scaling or (d) almost isotropic highlight with low scaling. A too large interval size leads either to (e) inconsistent anisotropy estimation with strong scaling or (f) almost isotropic highlight with low scaling. Too few samples lead either to (g) noise artifacts without filtering or (h) unreliable anisotropy estimation with a large filter size.

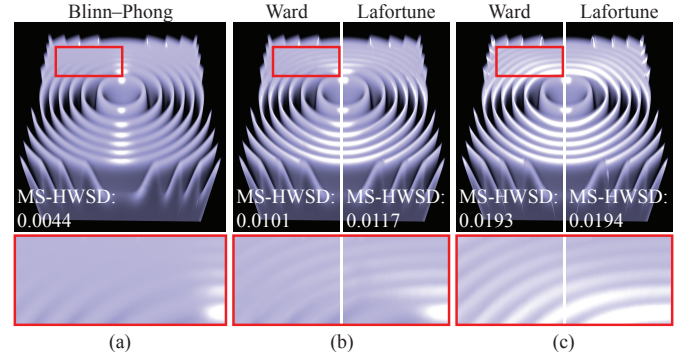


Fig. 7. Visualization of the test signal by Marschner and Lobb [33]. (a) Isotropic shading with Blinn-Phong’s BRDF. Anisotropic shading with Ward’s and Lafortune’s BRDF using (b) moderate scaling factors and (c) strong scaling factors for stretching the highlights.

estimation in Figure 6(e). In Figure 6(f), the inconsistency can be resolved by decreasing the scaling factors, but the highlights become isotropic. Although only 50 samples per voxel are necessary for noise-free results, this example demonstrates that the interval size needs to be chosen sufficiently small to faithfully detect anisotropy. For all further results of this paper, we employ a default value of $\varepsilon = 0.05$.

In Figures 6(g) and (h), we use again the settings of the reference in Figure 6(a), but we decrease the number of samples to $n = 10$. In Figure 6(g), we do not filter ($f = 1$) the covariance matrices, which leads to visible noise artifacts. In Figure 6(h), a large kernel of $f = 21$ voxels is employed for low-pass filtering. However, the specular highlight still exhibits low-frequency noise and its overall shape is irregular. We are omitting results for large radii, small interval sizes, and a high number of samples in Figure 6, as they do not provide additional insight.

In Figure 7, we study the impact of γ_w and γ_l for scaling the anisotropy for visualization using the test signal by Marschner and Lobb [33]. Although the ratio of the eigenvalues already provides a tool to automatically measure anisotropy, the magnitude of this ratio is often not adequate for direct use with a BRDF. Therefore, γ_w and γ_l act as global parameters for remapping the ratio to a more suitable range. For comparison, we show isotropic shading with Blinn-Phong’s BRDF in Figure 7(a) and the magnified region exhibits the typical band structures of the test signal. Figure 7(b) shows the same setup with Ward’s and Lafortune’s BRDF and moderate scaling factors of $\gamma_w = \gamma_l = 10$. The band structures are already better visible than with isotropic highlights. By further increasing the scaling factors to $\gamma_w = \gamma_l = 60$ in Figure 7(c), these features become even more salient. The difference between Ward’s and Lafortune’s BRDF is very small, but the comparison with Blinn-Phong’s BRDF shows the benefit of anisotropic shading.

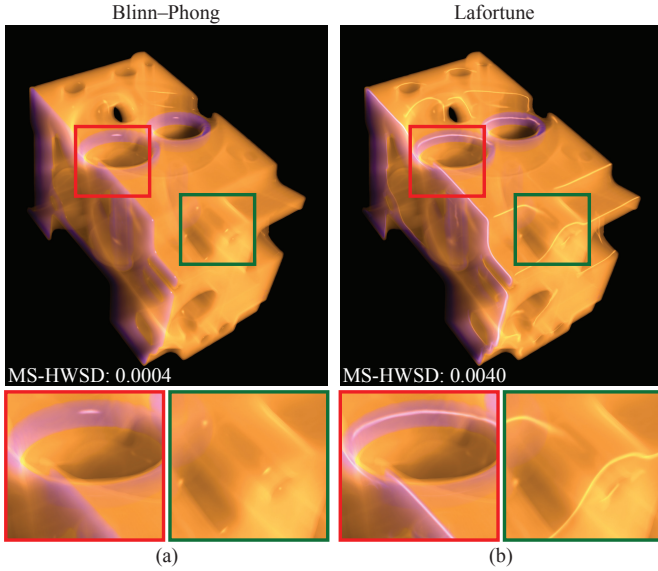


Fig. 8. Visualization of the Engine data set. (a) Isotropic shading with Blinn-Phong’s BRDF. (b) Anisotropic shading with Lafortune’s BRDF.

Our first real-world data set in Figure 1 shows one time step of a flow simulation with a resolution of 529^3 voxels using the λ_2 vortex criterion, a derived scalar field widely used for flow visualization of vortices. The high number of tubular structures and their complex spatial arrangement pose challenges for visualization. In Figure 1(a), we show the data set without any specular highlights using only the Henyey-Greenstein phase function for volumetric scattering. In Figure 1(b), isotropic highlights are added according to Blinn-Phong’s BRDF. Although the specular reflections contribute visual aids to convey the surface structures, their sparsity and non-uniform appearance can be misleading. In Figure 1(c), we employ our approach with anisotropic shading using Lafortune’s BRDF with an ambient radius of $r = 10$ voxels and $n = 100$ samples per voxel. In this case, the specular highlights are aligned with the tubular structures and they provide more visual cues due to their higher density and uniformity.

Figure 8 shows visualizations of the Engine data set with a resolution of $256 \times 256 \times 110$ voxels. The data set exhibits many isotropic surfaces combined with sharp edges and curved concavities. The high transparency and the faint shadows make it difficult to visually convey salient structures of the engine block. In Figure 8(a), we employ isotropic highlights using Blinn-Phong’s BRDF with high glossiness $\kappa = 80$. However, the highlights provide only few additional cues to help perceive the shape of the engine. In Figure 8(b), we employ a radius of $r = 7$ voxels and $n = 50$ samples per voxel to estimate anisotropy using Lafortune’s BRDF with the same high glossiness $\kappa_l = 80$. Furthermore, we use a moderate scaling factor of $\gamma_l = 10$ for visualization, which is sufficient to clearly accentuate the gaskets or the edges of the engine block as shown in the insets.

In Figure 9, the Mekanix data set is visualized with a resolution of $256 \times 256 \times 302$ voxels and the transfer function maps the skeleton to high opacity, while contextual structures of the torso are semi-transparent. In Figure 9(a), isotropic specular highlights due to Blinn-Phong’s BRDF are employed with low glossiness $\kappa = 10$. In contrast, Figure 9(b) shows the same setup with our anisotropic shading using Ward’s BRDF and visually comparable glossiness $\kappa_w = 0.2$. Furthermore, we employ a scaling factor of $\gamma_w = 40$ for visualization. Compared to isotropic shading, the bone structures like the ribs and the pelvis are visually more salient, especially in the areas where the semi-transparent tissue partially occludes them.

Figure 10 shows the Manix data set with a resolution of 512^3 voxels. In contrast to the previous data set, features of different scales are visualized, like parts of the skeleton, muscular tissue, and fine-veined blood vessels. In Figures 10(a) and (b), we employ high glossiness $\kappa = 30$ and $\kappa_l = 30$ for Blinn-Phong’s and Lafortune’s BRDFs, re-

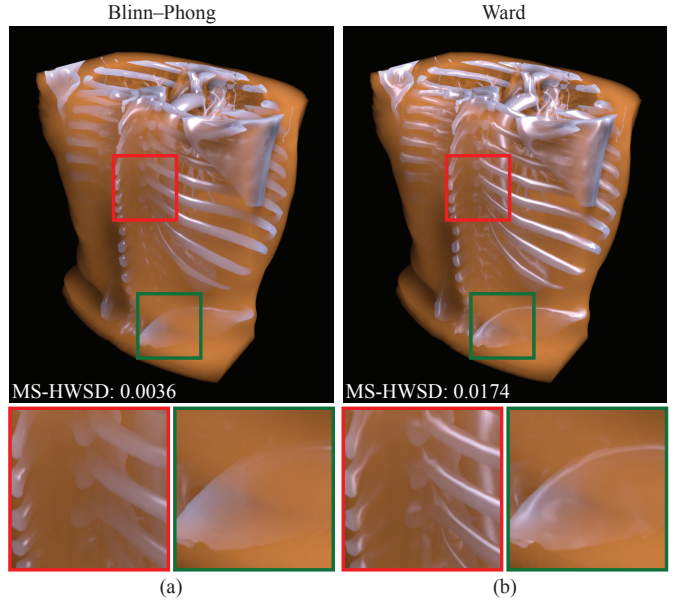


Fig. 9. Visualization of the Mekanix data set. (a) Isotropic shading with Blinn-Phong’s BRDF. (b) Anisotropic shading with Ward’s BRDF.

spectively. For anisotropic shading, we use a radius of $r = 30$ voxels, a number of $n = 100$ samples, and a scaling factor of $\gamma_l = 50$. The bluish region highlights parts of the cheek, exhibiting only weak anisotropy due to the almost uniform curvature. Our approach in Figure 10(b) automatically detects this local quasi-isotropic feature and its visualization is similar to isotropic shading in Figure 10(a). In contrast, the yellowish and greenish regions highlight areas of strong anisotropy, where our approach better reveals the fine structures of the auricle or the blood vessels. On a larger scale, the bone structures of the eye socket or the clavicle are also more salient. In Figures 10(c) and (d), we show the same comparison, but with low glossiness $\kappa = 5$ and $\kappa_l = 5$ for both BRDFs. Qualitatively, we observe similar findings: Although the difference between isotropic and anisotropic shading is less pronounced, the depiction of anisotropic structures is still improved.

In Table 1, we summarize performance measurements and memory consumptions. Depending on the data set, it is sufficient to employ $0.5 - 1.0 \times$ the resolution of the data set for the normals and the covariance matrices, as shown in the fourth column. We can see that memory consumption due to the covariance matrices is just as high as for the prefiltered normals. The computation time for the covariance matrices depends on the number of samples, which in turn usually depends on the radius and the complexity of the features. However, in simple cases like the ellipsoid, even very large radii do not require an excessive number of samples to obtain high quality. Rendering without any specular highlights always achieves the highest frame rates, but at the cost of few visual cues. Rendering with Blinn-Phong’s BRDF requires additional normal estimation, which is consistently slower, but the highlights also provide more structural information on shape. Anisotropic shading with Ward’s and Lafortune’s BRDF requires additional access to the covariance matrices, but we think that the gained benefit in our visual results compensates the slower performance in many cases. Finally, changing the transfer function or moving the light sources can be performed interactively in all cases and we do not provide additional measurements, because the only delay is due to shadow recomputation; however, this is not specific to our contribution.

In Table 2, we complement our performance measurements with scores of the MS-HWSD metric, focusing on results with high glossiness and high scaling factors. We employ volumetric scattering without any specular highlights as reference to compute the scores. The small absolute values of the scores are due to the fact that additional specular highlights usually change only a small number of pixels of the full image, especially for very high values of the glossiness

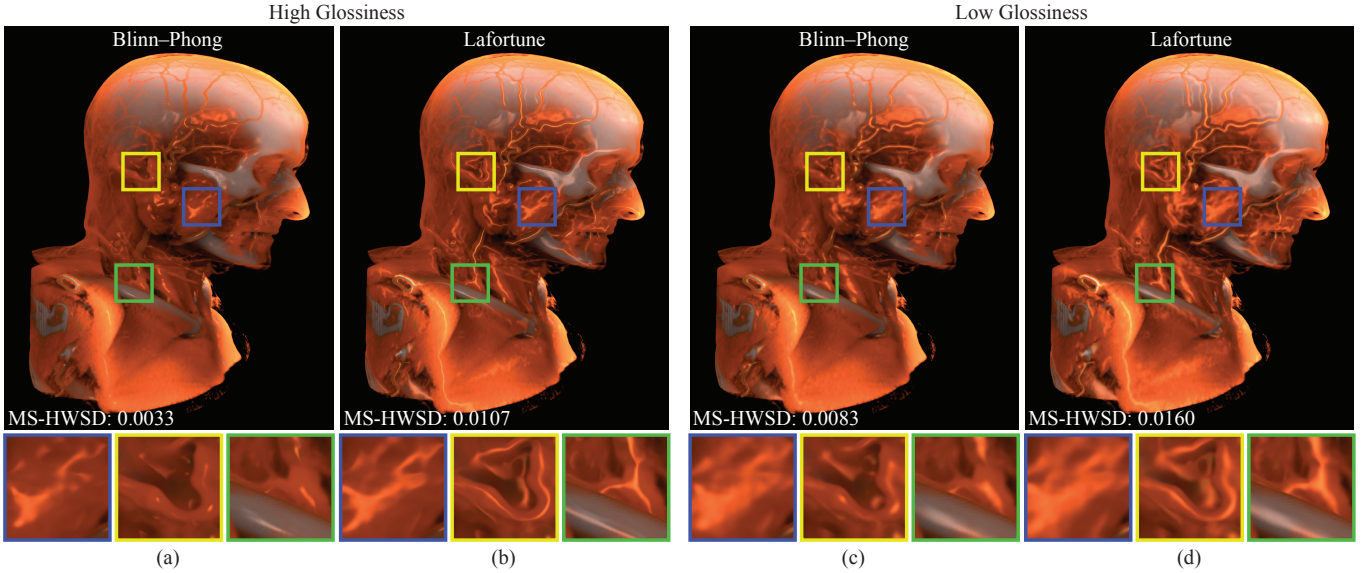


Fig. 10. Visualization of the Manix data. Shading with high glossiness using (a) Blinn-Phong’s isotropic BRDF and (b) Lafortune’s anisotropic BRDF. Shading with low glossiness using (c) Blinn-Phong’s isotropic BRDF and (d) Lafortune’s anisotropic BRDF.

Table 1. Overview of performance and memory consumption using a viewport size of 1024^2 pixels and early ray termination. The computation times for the normals N and the covariance matrices C are not included in the rendering performance. The resolution factors for N and C in the fourth column are with respect to the resolution of the data set. The interval size is $\varepsilon = 0.05$ for all results.

Data Set	Fig.	Res. Data	Res. N, C	n	r	N		C		Rendering [fps]			
						Mem. [MB]	Time [ms]	Mem. [MB]	Time [ms]	None	Bl.-Ph.	Ward	Lafortune
Vortex	1	$529 \times 529 \times 529$	$1/2$	100	10	53.2	6	53.2	1777	38	31	26	27
Ellipsoid	6	$512 \times 512 \times 512$	$1/2$	500	200	48.0	5	48.0	4575	71	62	50	50
Ma./Lobb	7	$512 \times 512 \times 512$	$3/4$	100	20	162.0	14	162.0	3775	83	73	52	54
Engine	8	$256 \times 256 \times 110$	1	50	7	20.6	2	20.6	627	97	88	68	69
Mecanix	9	$256 \times 256 \times 302$	1	100	30	56.6	6	56.6	2149	168	115	59	59
Manix	10	$512 \times 512 \times 512$	$3/4$	100	30	162.0	14	162.0	3609	104	80	63	64

as for the Engine data set, for example. In particular, Table 2 reveals that anisotropic shading consistently leads to higher scores than isotropic shading with Blinn-Phong’s BRDF, and the difference between Ward’s and Lafortune’s BRDF is negligible for our purposes. While in principle, even higher scores could be obtained by simply adding arbitrary structural information like noise to an image, our rendered results show that the additional cues due to our method visualize meaningful structures of a data set.

7 CONCLUSION

We have introduced a novel technique for anisotropic shading in DVR to improve the perception of complex surface-like structures. Compared to common isotropic shading, our anisotropic specular highlights are aligned with the anisotropy of the volumetric features, providing meaningful visual representatives to convey shape. Anisotropy is estimated solely from the data set, independent of the transfer function, which offers interactive data exploration without requiring recomputation. The memory footprint of our approach is twice as high as using prefiltered gradients with isotropic shading; however, we think that the additional benefit outweighs the higher resource consumption in many cases.

We have shown experimentally that our anisotropy estimation method with a small radius is closely related to the principal curvature directions. The advantage of our approach is, however, that the radius can be increased to avoid disturbing visual patterns or to highlight the more salient features of a data set. We have shown that the structural information is increased with anisotropic shading according to the MS-HWSD metric, but future work is necessary to investigate the perceptual benefits in more detail, for example, with a user-study like the one by Lindemann and Ropinski [30].

Table 2. Overview of the MS-HWSD scores using high glossiness and high scaling factors. All scores are with respect to volumetric scattering without any specular highlights.

Data Set \ BRDF	Vortex	Ellipsoid	Ma./Lobb	Engine	Mecanix	Manix
Blinn-Phong	0.0019	0.0565	0.0044	0.0004	0.0036	0.0033
Ward	0.0097	0.0966	0.0193	0.0041	0.0174	0.0109
Lafortune	0.0114	0.0961	0.0194	0.0040	0.0134	0.0107

The main limitation of our approach is the comparably long computation time for exploring the ambient region of a point to estimate anisotropy, which limits its use for quickly exploring time-dependent data sets. Here, future work could study more advanced sampling strategies to accelerate computation, for example, by deterministically exploring the isosurface at a given sample point. However, care has to be taken that a regular sampling pattern does not introduce a systematic bias in the anisotropy estimation.

Another drawback of our method is the number of parameters and the exploration of their values. Although we achieved good results with a constant interval size, the radius and number of samples strongly depend on the size of the features of interest. Future work could investigate algorithms that explore the different scales of a data set to automatically determine reasonable radii and sample densities. In particular, with an automatic algorithm, these parameters could be spatially adapted to implement an efficient and robust technique.

ACKNOWLEDGMENTS

The authors thank Andrea Beck, OsiriX, and all collaborators of <http://volvis.org> for providing data sets.

REFERENCES

- [1] E. H. Adelson. On seeing stuff: the perception of materials by humans and machines. In *Proc. of SPIE*, pages 1–12, 2001.
- [2] M. Ament, F. Sadlo, C. Dachsbacher, and D. Weiskopf. Low-pass filtered volumetric shadows. *IEEE Trans. Vis. Comput. Graph.*, 20(12):2437–2446, 2014.
- [3] M. Ament, F. Sadlo, and D. Weiskopf. Ambient volume scattering. *IEEE Trans. Vis. Comput. Graph.*, 19(12):2936–2945, 2013.
- [4] U. Behrens and R. Ratering. Adding shadows to a texture-based volume renderer. In *Proc. of the IEEE Symposium on Volume Visualization*, pages 39–46, 1998.
- [5] S. Bista, J. Zhuo, R. P. Gullapalli, and A. Varshney. Visualization of brain microstructure through spherical harmonics illumination of high fidelity spatio-angular fields. *IEEE Trans. Vis. Comput. Graph.*, 20(12):2516–2525, 2014.
- [6] J. F. Blinn. Models of light reflection for computer synthesized pictures. *SIGGRAPH Computer Graphics*, 11(2):192–198, 1977.
- [7] F. Caniard and R. W. Fleming. Distortion in 3d shape estimation with changes in illumination. In *Proc. of APGV*, pages 99–105, 2007.
- [8] S. Chandrasekhar. *Radiative Transfer*. Dover, 1960.
- [9] F. C. Crow. Summed-area tables for texture mapping. *SIGGRAPH Computer Graphics*, 18(3):207–212, 1984.
- [10] B. Csébfalvi and B. Domonkos. Prefiltered gradient reconstruction for volume rendering. In *Proc. of WSCG*, pages 49–56, 2009.
- [11] P. Dutré. Global illumination compendium. Technical report, KU Leuven, 2001.
- [12] J. A. Ferwerda, P. Shirley, S. N. Pattanaik, and D. P. Greenberg. A model of visual masking for computer graphics. In *Proc. of ACM SIGGRAPH*, pages 143–152, 1997.
- [13] R. W. Fleming, R. O. Dror, and E. H. Adelson. Real-world illumination and the perception of surface reflectance properties. *Journal of Vision*, 3:347–368, 2001.
- [14] M. Garland. *Quadric-based polygonal surface simplification*. PhD thesis, Georgia Institute of Technology, 1999.
- [15] A. Girshick, V. Interrante, S. Haker, and T. Lemoine. Line direction matters: An argument for the use of principal directions in 3d line drawings. In *Proc. of the International Symposium on Non-photorealistic Animation and Rendering*, pages 43–52, 2000.
- [16] M. Hadwiger, A. Kratz, C. Sigg, and K. Bühler. GPU-accelerated deep shadow maps for direct volume rendering. In *Proc. of the ACM SIGGRAPH/Eurographics Symposium on Graphics Hardware*, pages 49–52, 2006.
- [17] L. Henyey and J. Greenstein. Diffuse radiation in the galaxy. *Astrophysical Journal*, 93:70–83, 1941.
- [18] F. Hernell, P. Ljung, and A. Ynnerman. Local ambient occlusion in direct volume rendering. *IEEE Trans. Vis. Comput. Graph.*, 16(4):548–559, 2010.
- [19] Z. Hossain, U. Alim, and T. Möller. Toward high-quality gradient estimation on regular lattices. *IEEE Trans. Vis. Comput. Graph.*, 17(4):426–439, 2011.
- [20] V. Interrante. Illustrating surface shape in volume data via principal direction-driven 3d line integral convolution. In *Proc. of ACM SIGGRAPH*, pages 109–116, 1997.
- [21] D. Jönsson, J. Kronander, T. Ropinski, and A. Ynnerman. Historygrams: Enabling interactive global illumination in direct volume rendering using photon mapping. *IEEE Trans. Vis. Comput. Graph.*, 18(12):2364–2371, 2012.
- [22] D. Jönsson, E. Sundén, A. Ynnerman, and T. Ropinski. Interactive volume rendering with volumetric illumination. In *Eurographics STAR program*, pages 53–74, 2012.
- [23] G. Kindlmann and D. Weinstein. Hue-balls and lit-tensors for direct volume rendering of diffusion tensor fields. In *Proc. of IEEE Visualization*, pages 183–189, 1999.
- [24] G. Kindlmann, R. Whitaker, T. Tasdizen, and T. Möller. Curvature-based transfer functions for direct volume rendering: Methods and applications. In *Proc. of IEEE Visualization*, pages 513–520, 2003.
- [25] T. Kroes, F. H. Post, and C. P. Botha. Exposure render: An interactive photo-realistic volume rendering framework. *PLoS ONE*, 7(7):e38586, 2012.
- [26] J. Kronander, D. Jönsson, J. Low, P. Ljung, A. Ynnerman, and J. Unger. Efficient visibility encoding for dynamic illumination in direct volume rendering. *IEEE Trans. Vis. Comput. Graph.*, 18(3):447–462, 2012.
- [27] E. P. F. Lafortune, S.-C. Foo, K. E. Torrance, and D. P. Greenberg. Non-linear approximation of reflectance functions. In *Proc. of ACM SIGGRAPH*, pages 117–126, 1997.
- [28] M. Langer and H. Bülthoff. Depth discrimination from shading under diffuse lighting. *Perception*, 29(6):649–660, 2000.
- [29] M. Levoy. Display of surfaces from volume data. *IEEE Comput. Graph. Appl.*, 8(3):29–37, 1988.
- [30] F. Lindemann and T. Ropinski. About the influence of illumination models on image comprehension in direct volume rendering. *IEEE Trans. Vis. Comput. Graph.*, 17(12):1922–1931, 2011.
- [31] E. B. Lum, A. Stoppel, and K.-L. Ma. Using motion to illustrate static 3d shape-kinetic visualization. *IEEE Trans. Vis. Comput. Graph.*, 9(2):115–126, 2003.
- [32] O. Mallo, R. Peikert, C. Sigg, and F. Sadlo. Illuminated lines revisited. In *Proc. of IEEE Visualization*, pages 19–26, 2005.
- [33] S. R. Marschner and R. J. Lobb. An evaluation of reconstruction filters for volume rendering. In *Proceedings of IEEE Visualization*, pages 100–107, 1994.
- [34] N. Max. Optical models for direct volume rendering. *IEEE Trans. Vis. Comput. Graph.*, 1(2):99–108, 1995.
- [35] D. K. McAllister. *A Generalized Surface Appearance Representation for Computer Graphics*. PhD thesis, The University of North Carolina at Chapel Hill, 2002.
- [36] L. Neumann, B. Csébfalvi, A. Knig, and E. Gröller. Gradient estimation in volume data using 4d linear regression. *Comput. Graph. Forum*, 19(3):351–358, 2000.
- [37] D. Patel, V. Soltészová, J. M. Nordbotten, and S. Bruckner. Instant convolution shadows for volumetric detail mapping. *ACM Trans. Graph.*, 32(5):154:1–154:18, 2013.
- [38] M. Pauly, M. Gross, and L. P. Kobbelt. Efficient simplification of point-sampled surfaces. In *Proc. of IEEE Visualization*, pages 163–170, 2002.
- [39] B. T. Phong. Illumination for computer generated pictures. *Communications of the ACM*, 18(6):311–317, 1975.
- [40] B. Raymond, G. Guennebaud, P. Barla, R. Pacanowski, and X. Granier. Optimizing BRDF orientations for the manipulation of anisotropic high-lights. *Comput. Graph. Forum*, pages 313–321, 2014.
- [41] T. Ropinski, C. Döring, and C. Rezk-Salama. Interactive volumetric lighting simulating scattering and shadowing. In *Proc. of IEEE Pacific Visualization Symposium*, pages 169–176, 2010.
- [42] H. Sanftmann and D. Weiskopf. Illuminated 3d scatterplots. In *Proc. of the Eurographics / IEEE - VGTC Conference on Visualization*, pages 751–758, 2009.
- [43] P. Schlegel, M. Makhinya, and R. Pajarola. Extinction-based shading and illumination in GPU volume ray-casting. *IEEE Trans. Vis. Comput. Graph.*, 17(12):1795–1802, 2011.
- [44] G. Schussman and K.-L. Ma. Anisotropic volume rendering for extremely dense, thin line data. In *Proc. of IEEE Visualization*, pages 107–114, 2004.
- [45] I. Sobel. An isotropic 3x3x3 volume gradient operator. Technical report, Hewlett-Packard Laboratories, 1995.
- [46] S. Stegmaier, M. Strengert, T. Klein, and T. Ertl. A simple and flexible volume rendering framework for graphics-hardware-based raycasting. In *Proc. of the Eurographics / IEEE VGTC Conference on Volume Graphics*, pages 187–195, 2005.
- [47] Y. Tao, H. Lin, F. Dong, C. Wang, G. Clapworthy, and H. Bao. Structure-aware lighting design for volume visualization. *IEEE Trans. Vis. Comput. Graph.*, 18(12):2372–2381, 2012.
- [48] V. Soltészová, D. Patel, S. Bruckner, and I. Viola. A multidirectional occlusion shading model for direct volume rendering. *Comput. Graph. Forum*, 29(3):883–891, 2010.
- [49] Z. Wang, A. Bovik, H. Sheikh, and E. Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE Trans. Image Processing*, 13(4):600–612, 2004.
- [50] G. Ward. Measuring and modeling anisotropic reflection. *ACM SIGGRAPH Computer Graphics*, 26(2):265–272, 1992.
- [51] Y. Zhang and K.-L. Ma. Fast global illumination for interactive volume visualization. In *Proc. of the ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games*, pages 55–62, 2013.
- [52] M. Zöckeler, D. Stalling, and H.-C. Hege. Interactive visualization of 3d-vector fields using illuminated streamlines. In *Proc. of IEEE Visualization*, pages 107–113, 1996.