Ambient Volume Scattering

Marco Ament, *Student Member, IEEE*, Filip Sadlo, *Member, IEEE*,
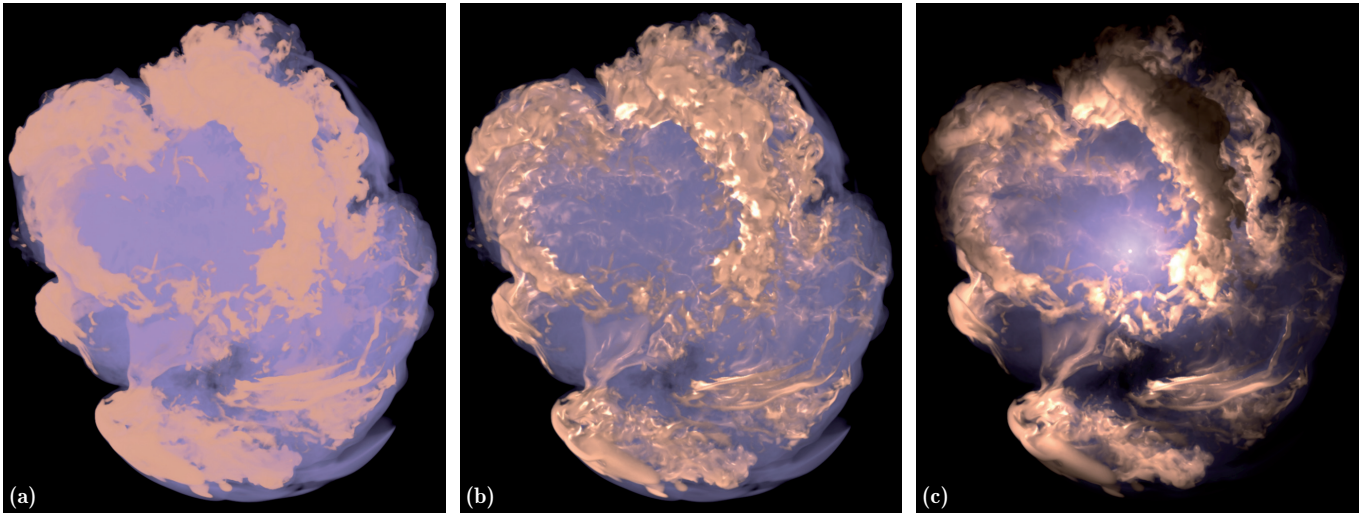and Daniel Weiskopf, *Member, IEEE Computer Society*

Fig. 1. Volume renderings of a supernova simulation with different optical models. (a) Standard emission–absorption model (64 fps). (b) Volumetric ambient occlusion (27 fps). (c) Our ambient scattering model with a light source in the center of the supernova (20 fps).

Abstract—We present ambient scattering as a preintegration method for scattering on mesoscopic scales in direct volume rendering. Far-range scattering effects usually provide negligible contributions to a given location due to the exponential attenuation with increasing distance. This motivates our approach to preintegrating multiple scattering within a finite spherical region around any given sample point. To this end, we solve the full light transport with a Monte-Carlo simulation within a set of spherical regions, where each region may have different material parameters regarding anisotropy and extinction. This precomputation is independent of the data set and the transfer function, and results in a small preintegration table. During rendering, the look-up table is accessed for each ray sample point with respect to the viewing direction, phase function, and material properties in the spherical neighborhood of the sample. Our rendering technique is efficient and versatile because it readily fits in existing ray marching algorithms and can be combined with local illumination and volumetric ambient occlusion. It provides interactive volumetric scattering and soft shadows, with interactive control of the transfer function, anisotropy parameter of the phase function, lighting conditions, and viewpoint. A GPU implementation demonstrates the benefits of ambient scattering for the visualization of different types of data sets, with respect to spatial perception, high-quality illumination, translucency, and rendering speed.

Index Terms—Direct volume rendering, volume illumination, ambient scattering, preintegrated light transport, gradient-free shading

✦

1 INTRODUCTION

Direct volume rendering (DVR) often relies on the optical model of emission and absorption [32]. However, spatial perception of shape can be difficult, since variation of luminance and shadows are missing, which provide important visual cues for spatial depth under natural lighting. The recent study by Lindemann and Ropinski [28] revealed that shadow-based volume rendering techniques are superior compared to local illumination to determine the relative size of volumetric features. A common alternative to directional shadows is volu-

metric ambient occlusion that samples the spherical neighborhood of each point to find occluders that attenuate ambient light creating local soft shadows. In this way, concavities appear darker than unoccluded regions. However, the study by Langer and Bülthoff [27] showed that human perception is not solely based on the "dark-means-deep model". In particular, complex interreflections influence the perception of shape under different lighting conditions. Our goal is to develop an illumination model that accounts for these aspects. We show how directional soft shadows can be combined with indirect lighting from multiple scattering. Similar to ambient occlusion, we focus on a finite spherical region around each point to compute ambient scattering based on physical light transport.

Many shading models [35] rely on normal vectors, which are usually derived from the gradients. However, in homogeneous subvolumes or in areas with low signal-to-noise ratio, the direction of the gradient is not numerically stable and shading is susceptible to artifacts. A common alternative is a single-scattering illumination model [32] based on shadow rays and a phase function. The drawback of that approach is that only a small fraction of light interacts with the medium, which can lead to strong darkening and hard shadows. The latter issues can be avoided with ambient occlusion techniques that integrate

---

- *Marco Ament is with VISUS, University of Stuttgart, Germany. E-mail: marco.ament@visus.uni-stuttgart.de.*
- *Filip Sadlo is with VISUS, University of Stuttgart, Germany. E-mail: sadlo@visus.uni-stuttgart.de.*
- *Daniel Weiskopf is with VISUS, University of Stuttgart, Germany. E-mail: weiskopf@visus.uni-stuttgart.de.*

extinction over a small spherical neighborhood [9, 16, 39, 41, 46] or a directional cone [44] around each point to compute a local occlusion factor for shading of ambient light.

More complex interreflections of light lead to a more natural illumination, but the computation of multiple scattering requires the integration of high-dimensional light transport in the vicinity of each point. Stochastic Monte-Carlo integration is well-suited to solve radiative transfer in participating media [4]. However, to obtain noise-free results, high sampling rates are required, which is computationally too expensive for interactive volume visualization. Nonetheless, high sampling rates are also a well-known issue in the context of the classic emission–absorption model, especially, if the transfer function contains high frequencies like sharp peaks to visualize isosurfaces. An established technique to accelerate rendering is the preintegration of the transfer function [10, 40] by means of a look-up table.

Adapting preintegration to volumetric scattering is, however, not straightforward because scattering requires the solution of global illumination. For inhomogeneous data sets, the dimension of any preintegration technique exceeds practical limits in terms of computation and storage. Therefore, it is vital to simplify light transport by taking the specific demands of interactive volume visualization into account. In contrast to photorealistic rendering, we do not solve illumination on a global scale, but we motivate an optical model on a mesoscopic scale that approximates illumination in the ambient volume of each point.

Our contribution is an optical model for ambient scattering by preintegrating light transport in spherical subvolumes by means of a Monte-Carlo simulation. An important property of our approach is that the rather expensive precomputation neither depends on a specific data set nor on the transfer function. Furthermore, the results can be stored efficiently in small look-up tables, which are well-suited for GPU rendering. We present a simple ray casting algorithm that employs our preintegration table and that is combinable with many other established volume rendering techniques, such as gradient-based shading and ambient occlusion. The benefit of our approach is an interactive visualization of volumetric structures that employs physically based multiple scattering. As a unique feature, we can control the full range of anisotropic scattering by means of the phase function. In this way, forward and backward scattering effects can be simulated in the ambient region of each sample. As a consequence, we can render subsurface scattering and translucency effects simultaneously. Scattering and soft shadows can be controlled with only two parameters: anisotropy and ambient radius. Our approach is well-suited for interactive volume exploration, even of time-dependent data.

2 RELATED WORK

Light transport in participating media is described by the radiative transfer equation (RTE) [5]. Max [32] presented an overview of derived optical models commonly used in DVR and he also emphasized the importance of illumination effects. Our approach relies on the preintegration of light transport in a spherical volume. In this spirit, we extend preintegration of emission and absorption [10, 40] with additional scattering. More recently, specialized variants of preintegration have been introduced by Knoll et al. [24] and by Ament et al. [1] for rendering isosurfaces and interval volumes, respectively. Lum et al. [30] presented an approximation to preintegrated lighting by linearly interpolating between two preintegration tables. Guetat et al. [11] further developed this approach with non-linear gradient interpolation. However, to the best of our knowledge, preintegration has not been applied to ambient scattering, before.

The classification of isosurfaces via shading parameters by means of a lighting transfer function was introduced by Lum and Ma [29]. An alternative to surface-based illumination models are gradient-free shading techniques like volumetric halos [3, 8]. In a similar way, we employ a spherical volume in the ambient region of each sample, but we compute full light transport with anisotropic scattering. Jönsson et al. [20] presented a comprehensive survey of advanced illumination techniques including scattering and shadows. In early work, Behrens and Ratering [2] rendered soft directional shadows by superimposing a shadow volume with the original scalar field. Zhang and

Crawfis [49] incorporated shadows into sheet-based splatting techniques; however, the approach is not ideally suitable for GPU processing. Data structures suitable for GPUs were presented by Hadwiger et al. [12], who adapted deep shadow maps to DVR. Kronander et al. [26] employed a truncated spherical harmonics expansion to approximate the directional visibility of each voxel of a multiresolution grid; however, the storage overhead can be significant.

Stewart [46] shaded isosurfaces by assuming diffuse ambient light that is occluded only in the vicinity of each surface point to improve perceptual cues. Ambient occlusion can be also employed for illustrative and saliency-guided volume rendering, as shown by Ruiz et al. [41]. Hernell et al. [15, 16] employ ray casting to compute an ambient attenuation factor in the spherical neighborhood of each voxel on a multi-resolution grid. Schott et al. [44] presented directional occlusion shading that does not require any precomputation but is restricted to a single head light. Ropinski et al. [39] precomputed clustered histograms of the surrounding scalar values of each voxel to obtain a vicinity representation that is independent of the transfer function. However, the generation and the clustering of the histograms depend on the data set and computation can take up to several hours. Schlegel et al. [43] exploited the fact that the computation of soft shadows and ambient occlusion does not depend strongly on the spatial distribution of the surrounding occluders, which allows the use of summed area tables (SAT) for fast summation. In contrast, we utilize SATs to determine the average extinction coefficient in a spherical neighborhood that we use to look-up preintegrated light transport.

One of the first approaches to interactive scattering in DVR was introduced by Kniss et al. [22, 23]. In their approach, forward-directed scattering is estimated inside a cone toward the light source by repeated blurring operations. This approach was extended to GPU-based ray casting by Ropinski et al. [38] by utilizing an illumination volume. Sundén et al. [47] employed plane sweeping in image space to render similar lighting effects with lower memory demands. However, common to these techniques is that the phase function is restricted to forward scattering. For the physically based rendering of reflection nebulae, Magnor et al. [31] presented a multi-resolution approach that combines scattering on multiple scales in image space. However, their approach is restricted to smoothly varying data and is not suitable for rendering of isosurfaces. Moon et al. [33] employed precomputed statistics of scattering to accelerate path tracing, but the method is limited to homogeneous media. The dipole method by Jensen et al. [18] is suitable to efficiently render subsurface scattering effects in a slab by exploiting a diffusion approximation [45]; however, the approach is not directly applicable for volume rendering. More recently, Zhang and Ma [50] presented a volume rendering technique that approximates global illumination with a convection–diffusion model by assuming an optically thick medium. The method achieves interactive performance, but it is not capable of rendering anisotropic scattering in optically thin media.

Since full global illumination of participating media is beyond the scope of this paper, we refer to the survey of Cerezo et al. [4]. We employ a Monte-Carlo simulation, based on path tracing [21, 42], for the preintegration of light transport. Csébfalvi and Szirmay-Kalos [7] employed Monte-Carlo integration for volume rendering, but the optical model did not involve scattering. Wyman et al. [48] precomputed global illumination for isosurface rendering using spherical harmonics; however, the method cannot incorporate volumetric light interaction. Salama [36] presented a Monte-Carlo algorithm that approximates volumetric light transport by restricting scattering events to a limited set of isosurfaces. Similarly, Kroes et al. [25] presented a hybrid Monte-Carlo algorithm that samples the bidirectional reflectance distribution function at surface-like structures and the phase function in volumetric regions. However, only single scattering is solved and interactivity heavily relies on progressive refinement. Recently, Jönsson et al. [19] have introduced an extension of the photon mapping algorithm [17]. By tracking the history of photon trajectories, a full recomputation of light transport can be avoided when the transfer function is changed, but is still necessary when the light source is moved or when time-dependent data is visualized.

3 VOLUMETRIC ILLUMINATION

Let us begin by discussing the principles of volumetric illumination. We first review the basics of global light transfer in Section 3.1, before we motivate and introduce our novel illumination model of ambient light transfer in Section 3.2.

3.1 Global Light Transfer

In a participating medium with extinction coefficient $\sigma_t(x)$, light transfer is described by the RTE [5]:

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

where $T(x_1, x_2)$ is the transmittance between two points x_1 and x_2:

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

In Eqn. (1), the radiance $L(x, \omega)$ at position x in direction ω is the sum of the attenuated background radiance $L_b(x_b, \omega)$ from background position x_b and the source radiance of the medium $L_m(x, \omega)$ integrated along the view ray:

$$L_m(x, \omega) = (1 - \Lambda) L_e(x, \omega) + \Lambda L_i(x, \omega), \quad (3)$$

where $\Lambda = \sigma_s / \sigma_t$ is the scattering albedo and σ_s denotes the scattering coefficient. Furthermore, the extinction coefficient can be written as $\sigma_t = \sigma_s + \sigma_a$, where σ_a is the absorption coefficient. In Eqn. (3), emissive radiance is denoted with $L_e(x, \omega)$ and in-scattered radiance $L_i(x, \omega)$ is given by:

$$L_i(x, \omega) = \frac{1}{4\pi} \int_{\Omega} P(\omega', \omega) L(x, \omega') \, d\omega', \quad (4)$$

where Ω denotes the sphere of all directions. The phase function $P(\omega', \omega)/4\pi$ is the probability density that radiance is scattered from incident direction ω' to direction ω.

3.2 Ambient Light Transfer

The computationally most expensive part of global illumination is the in-scattering term of Eqn. (4), since an integration over the entire sphere of directions is required and the solution $L(x, \omega)$ appears on the right-hand side again. In the emission–absorption model, in-scattering is neglected and light transfer reduces to the classic volume rendering equation, which can be solved efficiently with GPU-based ray casting, for example. Therefore, we seek to find an optical model that is based on a one-dimensional integral, but that approximates in-scattering at each point of an eye ray. The same principle was exploited in ambient occlusion before, but only to approximate local shadows.

The motivation of our approach is based on two important observations. First, in a participating medium, according to Eqn. (2), radiance is attenuated exponentially with growing distance due to absorption and out-scattering. Therefore, long-range scattering effects contribute only little energy to a sample, especially in optically thick media or in the presence of almost opaque isosurfaces. However, in the vicinity of a voxel, scattered light from all incident directions accumulates and contributes to anisotropic volumetric shading. For this reason, we compute multiple scattering only in the ambient region of each sample point, similar to the computation of shadows in ambient occlusion.

The second observation is that the computation and sampling techniques of ambient occlusion cannot be employed for ambient scattering, since the computation of light transport has infinite dimension and depends on the anisotropy of the phase function. In general, high-dimensional integrals can be solved well with Monte-Carlo integration; however, the error in the estimator decreases only at a rate of $O(n^{1/2})$, where n is the number of samples taken [34]. For interactive visualization, it follows that even for a finite ambient region around each point, it remains too expensive to compute light transport in each frame. As a consequence, we opt for a Monte-Carlo preintegration by means of volumetric path tracing.
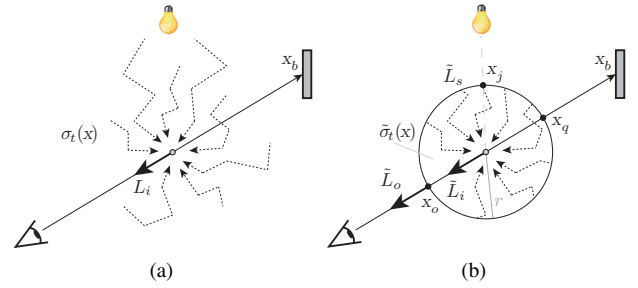

Fig. 2. (a) In-scattered radiance L_i on a global scale. (b) In-scattered ambient radiance $\tilde{L}_i$ on a mesoscopic scale. Scattering effects are restricted to a spherical region S_r of radius r with ambient extinction coefficient $\tilde{\sigma}_t$ and boundary condition $\tilde{L}_s$ on the surface. Moreover, the radially outgoing radiance $\tilde{L}_o$ integrates the in-scattered radiance $\tilde{L}_i$ along the line of sight inside the sphere.

3.2.1 Ambient Scattering

In general, the dimension of the parameter space of precomputed light transfer in a spherical volume is far too high. The spatially varying distribution $\sigma_t(x)$ prevents a straightforward approach, since, even for 8-bit data sets, the number of possible combinations is 256^m, when m is the number of voxels inside the sphere S_r of radius r as shown in Figure 2. However, previous work on ambient occlusion [39, 43] showed that the spatial distribution of the extinction coefficient inside the sphere does not heavily influence the result. We exploit and adapt this observation to ambient scattering by substituting:

$$\sigma_t(x) \rightarrow \tilde{\sigma}_t(x) = \frac{1}{\frac{4}{3}\pi r^3} \int_V \sigma_t(x') \, dV, \quad (5)$$

where $x' \in S_r(x)$ is a point inside the sphere of volume V with center x and $\tilde{\sigma}_t$ is the average extinction coefficient inside $S_r(x)$. We denote $\tilde{\sigma}_t$ as ambient extinction coefficient. Since $S_r(x)$ encloses different regions of the volume at each point, $\tilde{\sigma}_t(x)$ also depends on the location x. With this simplification, only a single dimension is required in terms of the extinction coefficient. Furthermore, we can preintegrate light transfer for a large range of values, which makes our approach independent from a specific data set or a specific transfer function. We can now approximate in-scattering of Eqn. (4) with the following term:

$$L_i(x, \omega) \approx \tilde{L}_i(x, \omega) = \frac{1}{4\pi} \int_{\Omega} P(\omega', \omega) \tilde{L}(x, \omega') \, d\omega', \quad (6)$$

where $\tilde{L}(x, \omega)$ is the radiance contribution due to ambient scattering inside the sphere $S_r(x)$. The outgoing radiance $\tilde{L}_o(x_o, \omega_o)$ that leaves the sphere at location x_o in radial direction ω_o is then described by:

$$\tilde{L}_o(x_o, \omega_o) = \tilde{T}(x_q, x_o) \tilde{L}_s(x_q, \omega_o) + \int_{x_q}^{x_o} \tilde{T}(x', x_o) \tilde{\sigma}_t(x') \tilde{L}_m(x', \omega_o) \, dx',$$
$$(7)$$

where x_q denotes the exit point of the eye ray and $\tilde{L}_s(x_q, \omega_o)$ is the corresponding boundary condition on the sphere surface that we describe later in more detail. The source radiance of the ambient medium becomes:

$$\tilde{L}_m(x, \omega) = (1 - \tilde{\Lambda}) \tilde{L}_e(x, \omega) + \tilde{\Lambda} \tilde{L}_i(x, \omega), \quad (8)$$

where $\tilde{\Lambda} = \tilde{\sigma}_s / \tilde{\sigma}_t$ is the ambient albedo. Similar to Eqn. (2), the ambient transmittance is given by:

$$\tilde{T}(x_1, x_2) = e^{-\int_{x_1}^{x_2} \tilde{\sigma}_t(x') \, dx'}, \quad (9)$$

for two points x_1 and x_2 inside of S_r. With Eqns. (6)–(9), we have a formal model to approximate multiple scattering in the vicinity of each sample point.
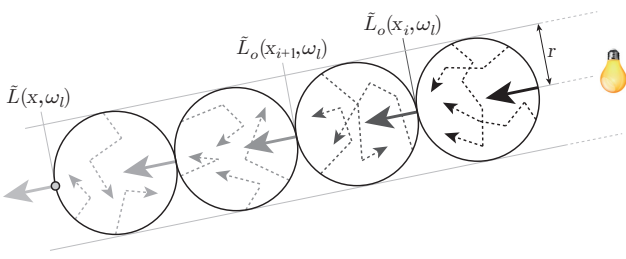
Fig. 3. Mesoscopic light transport with ambient scattering for soft shadows and translucency effects. Emitted radiance from the light source is propagated iteratively inside a tube of radius r in direction ω_l toward point x. The radially outgoing radiance $\tilde{L}_o(x_i, \omega_l)$ at step i serves as input for the computation of $\tilde{L}_o(x_{i+1}, \omega_l)$ at step $i+1$.

3.2.2 Soft Shadows with Tube Marching

With the previous model of ambient scattering, we can further extend our approach with soft shadows and translucency effects, like shown in Figure 3. In the following, we assume that $\tilde{L}_o$ can be computed efficiently for any point and any direction in the volume, which will be shown later on. By assuming a piece-wise constant $\tilde{\sigma}_t$, we can trace a ray with large step size $2r$ from the light sources x_l to any point x in the volume to accumulate the effects of shadowing and ambient scattering inside a tube of radius r. In this way, we can achieve light transport on a mesoscopic scale. The incident radiance $L(x, \omega_l)$ at point x in direction ω_l is obtained by a feedback algorithm that iteratively employs the result of step i as the input for step $i+1$. In particular, the radiance $\tilde{L}_o(x_i, \omega_l)$ serves as boundary condition to compute $\tilde{L}_o(x_{i+1}, \omega_l)$, which we explain later in more detail. With this model, soft shadows and translucency can be controlled by the anisotropy parameter of the phase function and by the radius.

4 THE ALGORITHM

Our overall algorithm consists of two major parts. First, full light transport is preintegrated for a range of parameter values. In Section 4.1, we provide details of the parameter space and show how to solve light transport by means of path tracing. This part of the algorithm is computationally expensive, but since the computation is independent of the data set, transfer function, lighting conditions, and viewpoint, it has to be performed only once. Apart from this step, no preprocessing is required for the rest of the algorithm.

The second major part is interactive volume rendering. In Section 4.2, we describe a technique that employs our preintegrated light transport to simulate anisotropic scattering and soft shadows. All our rendering steps are suited for interactive manipulation of the viewpoint, transfer function, phase function, and lighting conditions.

4.1 Preintegrated Light Transport

Based on our model of ambient light transfer in Section 3.2, we preintegrate light transport inside a homogeneous spherical volume S_r. In particular, we are interested in the radial radiance distribution $\tilde{L}_o$ that leaves the surface of the sphere given that the incoming radiance distribution can be provided as a boundary condition. Since we focus on highly scattering materials, we assume a constant high value of the albedo throughout the paper. Therefore, the absorption and scattering coefficients only depend on the extinction coefficient.

4.1.1 The Parameter Space

The remaining degrees of freedom for the preintegration are the radius r, ambient extinction coefficient $\tilde{\sigma}_t$, phase function P, boundary condition $\tilde{L}_s$, and an explicit dependency of the viewing angle.

However, we can reduce the parameter space by observing that the scale of r and the ambient extinction coefficient $\tilde{\sigma}_t$ are not independent. According to Eqn. (9), the transmittance throughout a homogeneous sphere of radius r is described by $e^{-r\tilde{\sigma}_t}$. Similarly, we get
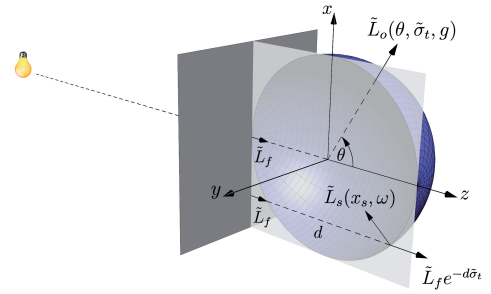

Fig. 4. Spherical geometry for preintegration of light transport. The dark shaded plane denotes a light front of approximately constant radiance $\tilde{L}_f$ from a far distant light source. The boundary condition $\tilde{L}_s(x_s, \omega)$ on the sphere surface is derived from $\tilde{L}_f$. Due to symmetry around the local z-axis, $\tilde{L}_o$ depends only on θ and the medium inside the sphere.

$e^{-kr\tilde{\sigma}_t}$ for a sphere of radius kr, which is equal to the transmittance for a sphere of radius r but with an extinction different by a factor k. Therefore, we can restrict our preintegration geometry to the unit radius and vary only $\tilde{\sigma}_t \in [0, \tilde{\Sigma}_t]$ for a sufficiently large range of values.

The second reduction concerns the phase function. In volume rendering, phase functions are usually symmetric around the incident direction [4, 19]; hence, they can be parameterized by the angle $\alpha = \arccos(\omega' \cdot \omega)$ between the incoming direction ω' and the outgoing direction ω. We follow common practice and employ the Henyey-Greenstein (HG) phase function [14] for the remainder of this paper although any other symmetric function could be used as well. With the HG phase function, the predominant scattering direction is controlled by the anisotropy parameter $g \in [-1, 1]$ that models a continuous range from backward to forward scattering:

$$P_{\text{HG}}(\alpha, g) = \frac{1 - g^2}{\left(1 + g^2 - 2g\cos\alpha\right)^{3/2}}. \quad (10)$$

The boundary condition $\tilde{L}_s$ uniquely determines how each spherical volume is illuminated from outside. Figure 4 illustrates how we setup the boundary condition. We create a local coordinate system in the center of a unit sphere so that the local z-axis is aligned with the incident direction of light. With a light source at a far distant location $(0, 0, -\infty)$, we can assume that light reaches the sphere in parallel at $(x, y, -r)$. Furthermore, we assume that the incoming radiance at the light front varies smoothly due to previous scattering events. Therefore, we model the incoming light front with a constant radiance value $\tilde{L}_f(x_f, \omega_z) = 1$ for each point x_f on the light front in direction $\omega_z = (0, 0, 1)$. In this way, we can later modulate the intensity of incoming light with any other value by means of a simple multiplication. The boundary condition $\tilde{L}_s(x_s, \omega)$ for any point x_s on the sphere in any direction ω can then be computed as:

$$\tilde{L}_s(x_s, \omega) = \tilde{L}_f e^{-d\tilde{\sigma}_t} \frac{1}{4\pi} P_{\text{HG}}(\arccos(\omega \cdot \omega_z), g), \quad (11)$$

where d is the distance that light travels through the sphere. A symmetric phase function further allows one to reduce the storage costs, since the radial radiance distribution on the surface of the sphere is symmetric around the local z-axis. Therefore, the view dependency reduces to the polar angle θ. It follows that storage costs can be reduced from a 3D spherical distribution to a 2D semicircle distribution that varies with $\theta \in [0, \pi]$.

With the previously discussed assumptions, the parameter space of our preintegration consists of three independent dimensions: angular dependency $\theta \in [0, \pi]$, ambient extinction $\tilde{\sigma}_t \in [0, \tilde{\Sigma}_t]$, and anisotropy parameter $g \in [-1, 1]$. The main advantages of this parameter space are its rather low dimension and its independence from a specific data set or transfer function. An appropriate upper limit for $\tilde{\Sigma}_t$ can be estimated easily from the spatial dimensions of typical data sets together with the desired maximum opacity value.

4.1.2 Volumetric Path Tracing

We employ unidirectional path tracing for participating media [42] to solve light transport and to compute the outgoing radial radiance $\tilde{L}_o$. We discretize the range of our 3D parameter space to obtain a finite set of values for which we preintegrate a solution. The angular discretization provides the starting points on the sphere surface for path tracing. Due to the previously discussed symmetry, all starting points lie in the *xz*-plane. Since we require only radial contributions of outgoing radiance, the initial direction of each path points toward the origin. In this way, Monte-Carlo integration is performed by looping over all discrete 3-tuples and by tracing *n* path samples for each tuple.

Each path sample requires a random walk through the unit sphere. For sampling the free path distance d, we employ importance sampling [34] of the transmittance term (9) with homogeneous ambient extinction $\tilde{\sigma}_t$. Given a uniformly distributed random number $\xi \in \,]0,1]$, the free path distance can be sampled with $d = -\ln \xi / \tilde{\sigma}_t$. At each interaction event, Russian roulette is used to determine if the path is terminated due to absorption. We follow common practice and employ the albedo to terminate the path if $\xi > \Lambda$. In case of a scattering event, the new direction of a path is determined by importance sampling the phase function. For details on importance sampling of the HG phase function, we refer the reader to the textbook of Pharr and Humphreys [34]. After the new direction is determined, the entire process is repeated by sampling again the free path distance. In this way, each path is traced until it is either terminated or intersects the surface of the sphere. In the latter case, the boundary condition is evaluated and radiance is transported according to the path throughput. After path tracing is finished for all discrete 3-tuples, $\tilde{L}_o(\theta, \tilde{\sigma}_t, g)$ is stored permanently in a 3D table. Figure 5 illustrates three slices of our table with an albedo of $\Lambda = 0.9$.

4.2 Interactive Volume Rendering

In this section, we describe our volume rendering algorithm. In the following, we assume that volume data is stored in the form of a discrete Cartesian lattice that samples a scalar field $s : \mathbb{R}^3 \rightarrow \mathbb{R}$ and the transfer function maps these data values to material properties such as extinction coefficient and color. We assume a constant high value of the albedo, since otherwise, light transport would be dominated by absorption. Our rendering approach consists of three major steps: First, for each voxel, a spherical neighborhood is scanned to determine the ambient extinction coefficient. Second, for soft shadows and translucency, light is distributed from the light sources within tubes of radius r by taking advantage of our preintegration tables. Third, for the final image, ray casting is employed to gather view-dependent radiance due to anisotropic scattering.

4.2.1 Ambient Extinction Volume

For reading the elements of our preintegration table, we require the ambient extinction coefficient $\tilde{\sigma}_t$ at each point of the volume. According to our model of light transfer in Section 3.2, we compute the ambient extinction by averaging σ_t over the local sphere $S_r(x)$ of radius r at each point x. In a discrete setting, $S_r(x)$ contains m voxels and Eqn. (5) becomes:

$$\tilde{\sigma}_t(x) \approx \frac{1}{V_s} \sum_{i=1}^{m} \sigma_t(x_i') \Delta V, \qquad x_i' \in S_r(x), \qquad (12)$$

where V_s is the volume of the discretized sphere and ΔV is the volume of one cell. Since σ_t depends on the transfer function, this computation is critical for efficient updates to facilitate interactive data exploration. Therefore, it is vital to accelerate this process as much as possible. Since the summation in Eqn. (12) is independent from a specific traversal order, we can employ summed area tables (SAT) [6], which can be implemented efficiently on the GPU [13]. However, the evaluation of volumetric SATs is restricted to cuboid-shaped regions. Therefore, we approximate the sphere $S_r(x)$ with a cube $C_r(x)$ of volume V_s and we get:

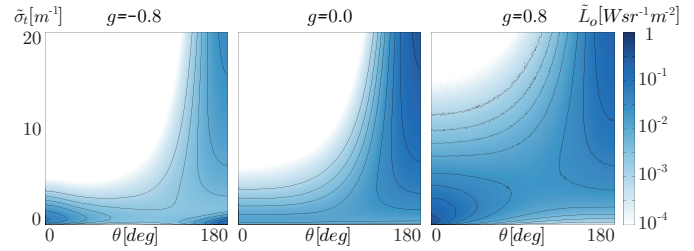$$\tilde{\sigma}_t(x) \approx \frac{1}{V_s} \mathrm{SAT}(C_r(x)). \qquad (13)$$

Fig. 5. Three slices of the preintegration table with an albedo of $\Lambda = 0.9$. Radiance $\tilde{L}_0$ is plotted logarithmically over the polar angle θ and the ambient extinction coefficient $\tilde{\sigma}_t$. From left to right, anisotropy of the HG phase function varies from backward scattering $g = -0.8$ over isotropic scattering $g = 0$ to forward scattering $g = 0.8$.

In contrast to previous work on DVR with summed area tables [9, 43], we do not derive an ambient occlusion factor from this computation, but we employ the SAT to obtain an ambient extinction value for the look-up in our preintegration table for ambient scattering. The main advantages of using SATs are the fast computation and the constant-time evaluation of cubical regions of any size. In this way, the ambient scattering radius r can be changed easily and recomputation time only depends on the data set size. Since the ambient extinction volume remains constant for camera or light source movement, it can be cached in a Cartesian grid, similar to an ambient occlusion volume.

4.2.2 Distribution of Light

In the next step, radiance is distributed from all light sources to implement our model of mesoscopic light transport by means of shadow-and-scatter tubes, like illustrated in Figure 3. In this spirit, our approach is similar to previous techniques that employed light cones [23, 38, 43, 44] to approximate advanced illumination. However, we employ our preintegration tables to render soft shadows and translucency by means of multiple scattering according to the anisotropy of the phase function.

In the preintegration step, we assumed $\tilde{L}_f = 1$ to compute the boundary condition in Eqn. (11). Since every table element $\tilde{L}_o$ is proportional to $\tilde{L}_f$, we can modulate the preintegrated results by an arbitrary factor to obtain the result for any value of $\tilde{L}_f$ and any radius r:

$$\tilde{L}_o(\theta, \tilde{\sigma}_t, g, \tilde{L}_f, r) = \tilde{L}_f \cdot \tilde{L}_o(\theta, r\tilde{\sigma}_t, g). \qquad (14)$$

We exploit Eqn. (14) to implement a feedback loop for a tube marching algorithm. Starting with radiance L_l from the light source, we define the following iteration procedure:

$$\tilde{L}_f(x_0, \omega_l) := L_l \qquad (15)$$

$$\tilde{L}_f(x_i, \omega_l) := \tilde{L}_f(x_{i-1}, \omega_l) \cdot \tilde{L}_o(0, r\tilde{\sigma}_t(x_{i-1}), g), \qquad (16)$$

for $i = 1, \ldots, k+1$ with $\theta = 0$. To avoid banding artifacts for large steps, we compute radiance values at positions x_k and x_{k+1}, where the point x of computation lies in between. The distributed radiance is then obtained by linear interpolation:

$$\tilde{L}_d(x, \omega_l) = (1-\eta)\tilde{L}_f(x_k, \omega_l) + \eta \tilde{L}_f(x_{k+1}, \omega_l), \qquad (17)$$

where $\eta = \|x - x_k\| / \|x_{k+1} - x_k\|$. In addition to realistic illumination effects, the usage of preintegrated light transport allows us to take large steps of $\Delta x = \|x_i - x_{i-1}\| = 2r$ for tube marching compared to traditional shadow ray marching in a single scattering model. An efficient implementation of the distribution of light is crucial for interactive data exploration, since recomputation is necessary as soon as the transfer function, phase function, or lighting conditions change. Similar to the ambient extinction volume, $\tilde{L}_d$ can be stored optionally in a radiance cache.

4.2.3 Ambient Ray Casting

The last step of our rendering algorithm is ray casting from the camera. The previously distributed radiance is gathered to achieve anisotropic shading by evaluating the preintegration table with the view-dependent angle θ. We solve for the radiance at the camera location x_c by approximating Eqn. (1) with:

$$L(x_c, \omega) \approx T(x_b, x_c)L_b(x_b, \omega) + J_e(x_b, x_c, \omega) + \tilde{J}_i(x_b, x_c, \omega). \quad (18)$$

We substitute $\sigma_a = \sigma_t(1 - \Lambda)$ and $\sigma_s = \sigma_t\Lambda$ and we define:

$$J_e(x_b, x_c, \omega) := \int_{x_b}^{x_c} T(x', x_c)\sigma_a(x')L_e(x', \omega)\,dx', \quad (19)$$

$$\tilde{J}_i(x_b, x_c, \omega) := \int_{x_b}^{x_c} T(x_o, x_c)\sigma_s(x')\tilde{L}(x', \omega, r)\,dx'. \quad (20)$$

In Eqn. (19), J_e describes the integrated emission–absorption term, which can be solved with standard DVR. In Eqn. (20), $\tilde{J}_i$ is the approximated term due to ambient in-scattering. Note that the transmittance is computed from the outgoing position $x_o = x' + r\omega$ to the camera x_c to avoid multiple attenuation inside the sphere (see also Figure 2(b)). The term $\tilde{L}(x', \omega, r)$ describes the sum of all distributed and in-scattered radiance from all l light sources:

$$\tilde{L}(x', \omega, r) = \sum_{j=1}^{l} \tilde{L}_{d,j}(x_j)\tilde{L}_o(\theta_j, r\tilde{\sigma}_t(x'), g), \quad (21)$$

where x_j is the intersection point of the j-th shadow ray and the sphere $S_r(x')$. The previously distributed radiance $\tilde{L}_{d,j}$ serves again as boundary condition for the modulated look-up of $\tilde{L}_o(\theta_j, r\tilde{\sigma}_t(x'), g)$ in the preintegration table. The corresponding angle is obtained by $\theta_j = \arccos(\omega \cdot \omega_j)$, where ω_j is the incident direction of the j-th shadow ray. By discretizing the integrals in Eqns. (19) and (20) with finite Riemann sums, ray marching can be employed for rendering.

5 IMPLEMENTATION

Our implementation consists of two independent parts. We implemented preintegration of ambient light transfer with path tracing on the CPU using standard C++. We employ a uniform spacing for the discretization of the parameter space to compute the radiance distribution L_o for a finite set of values. The results are stored permanently on disk for independent usage at rendering.

For comparison, we also implemented standard DVR using an emission–absorption model. In addition, since our model focuses on light transport in an ambient region, we also implemented SAT-based ambient occlusion [43] to compare the optical properties. Furthermore, we implemented a single scattering model with shadow rays to demonstrate the visual impact of multiple scattering. Although our model does not depend on normal vectors, we demonstrate that our approach is combinable with gradient-based shading by optionally adding specular highlights. In addition, dark regions can be illuminated optionally with SAT-based ambient occlusion.

We implemented all rendering algorithms on the GPU using CUDA. We employ 3D textures for storing the data set, the preintegration table, and for caching the ambient extinction volume $\tilde{\sigma}_t$ as well as the distributed radiance L_d. For the single scattering model, we also cache the transmittance T in a 3D texture. When the transfer function changes, all caches are recomputed. For our ambient extinction volume and ambient occlusion, we implemented the double-buffering algorithm of Hensley et al. [13] to compute the SAT on the GPU. The final image of all optical models is rendered with ray casting and early ray termination. The pseudocode of ambient scattering that solves Eqn. (20) is illustrated in Algorithm 1.

6 RESULTS

We discuss the visual quality and the performance of our approach compared to several common techniques in DVR with six different

---

Algorithm 1 Computation of $L(x_c, \omega)$ (without emission).

Require: camera position x_c, view direction $\omega_v = -\omega$, ambient radius r, step size Δt, albedo Λ, anisotropy g, background radiance L_b

1: $\tilde{J}_i, \tau = 0; T = 1;$
2: **if** intersectVolume($x_c, \omega_v, t_{near}, t_{far}$) **then**
3:    $t = t_{near};$
4:    **while** $t \leq t_{far}$ **do**
5:       $x' = $ getRayPosition(x_c, ω_v, t);
6:       $x_o = x' - r \cdot \omega_v;$
7:       $(C_{rgb}, \sigma_t) = $ sampleTransferFunction(x');
8:       $\sigma_s = C_{rgb} \cdot \sigma_t \cdot \Lambda;$
9:       $\tilde{\sigma}_t = $ sampleAmbientExtinctionVolume(x');
10:      $\tilde{L} = 0;$
11:      **for all** activeLightSources l_j **do**
12:         $\omega_j = $ normalize($x' - $ getLightPosition(l_j));
13:         $x_j = x' - r \cdot \omega_j;$
14:         $\theta_j = \arccos(\text{dot}(-\omega_v, \omega_j));$
15:         $\tilde{L}_o = $ samplePreintegrationTable($\theta_j, r \cdot \tilde{\sigma}_t, g$);
16:         $\tilde{L}_{d,j} = $ sampleRadianceCache(x_j, l_j);
17:         $\tilde{L} = \tilde{L} + \tilde{L}_{d,j} \cdot \tilde{L}_o;$
18:      **end for**
19:      $\tilde{J}_i = \tilde{J}_i + T \cdot \sigma_s \cdot \tilde{L} \cdot \Delta t;$
20:      $(C_{rgb}, \sigma_t) = $ sampleTransferFunction(x_o);
21:      $\tau = \tau + \sigma_t \cdot \Delta t;$
22:      $T = \exp(-\tau);$
23:      $t = t + \Delta t;$
24:    **end while**
25:    **return** $T \cdot L_b + \tilde{J}_i;$
26: **end if**

---

data sets from astronomy[1], medicine[2,3], and computational fluid dynamics (CFD). All results were obtained with an Intel Core i7 2.8 GHz CPU, 8 GB RAM, and an NVIDIA GTX-560 GPU.

The memory footprint of our method is comparable with previous volume illumination techniques. Similar to Schlegel et al. [43], we require a SAT, but only as a temporary data structure to efficiently compute the ambient extinction volume, which is then stored in a 3D texture with the same resolution as the data set. Optionally, we employ an additional 3D texture for the radiance cache [38] to accelerate rendering; however, since multiple scattering casts soft shadows, we employ a reduced cache of only half the resolution of the data set for all results in this paper without any noticeable difference. For larger data sets, a radiance cache becomes infeasible and illumination needs to be recomputed in each frame. Specific to our method is the storage of the preintegration table. We employ the same table for all data sets with an albedo of $\Lambda = 0.9$ and a resolution of $256 \times 256 \times 19$, which uniformly samples the ranges $\theta \in [0, \pi]$, $\tilde{\sigma}_t \in [0, 20]$, and $g \in [-0.9, 0.9]$; hence, with 32-bit floating point data, the table requires only 4.75 MB of memory, independent of the data set size.

Figure 1 shows volume renderings of a supernova simulation. In Figure 1(a), the standard emission–absorption model is employed, which achieves the highest performance but cannot visualize the fine filaments of the outer layer. With volumetric ambient occlusion, in Figure 1(b), more details emerge due to local shadows and gradient-based shading, but comprehension of spatial depth is still difficult. In Figure 1(c), a single light source in the center illuminates the volumetric features with ambient scattering $g = 0.6$, while rendering performance and delays from transfer function updates are still comparable with ambient occlusion, according to Table 1. Furthermore, changing the anisotropy or moving the light source can be done interactively.

Figure 5 illustrates our preintegration table for three different anisotropy values. The Monte-Carlo simulation took about 2–20 h using 1–10 k path samples with a single-threaded non-optimized CPU implementation. We plot radiance logarithmically over the polar angle θ and the ambient extinction value $\tilde{\sigma}_t$. In Figure 5(a), strong backward scattering reflects light mainly in the backward direction (large θ) for optically thin media (small $\tilde{\sigma}_t$). However, due to repeated back scattering, a substantial amount of energy is also transported in the forward direction (small θ). With isotropic scattering, radiance is distributed evenly in all directions for thin media, but for thick media (large $\tilde{\sigma}_t$), radiance is no longer transported but mainly reflected back-

---

[1]http://vis.cs.ucdavis.edu/VisFiles/pages/supernova.php
[2]http://www.nlm.nih.gov/research/visible/
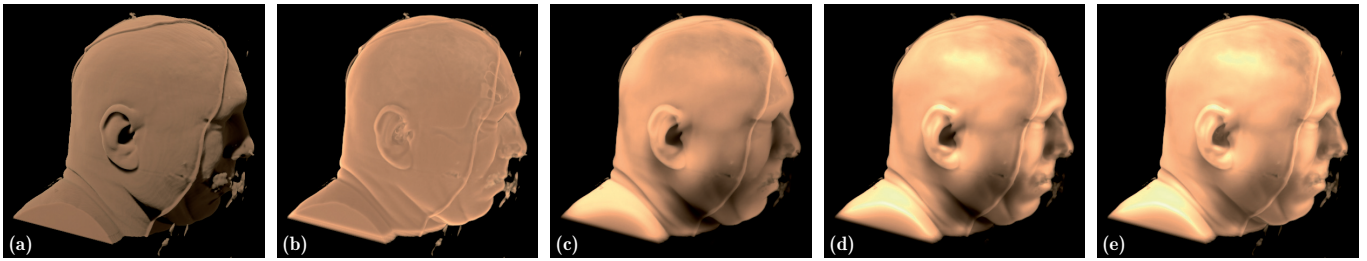[3]http://www.osirix-viewer.com/datasets/

Fig. 6. Visualization of the Visible Human data set with different optical models. (a) Single scattering model with isotropic scattering $g = 0.0$ using shadow rays. (b) Volumetric ambient occlusion with a radius of $r = 14$ voxels. (c) The same lighting conditions as in (a), but with ambient scattering $g = 0.0$ and a radius of $r = 14$ voxels. (d) Additional specular highlights with gradient-based shading. (e) Additional light with ambient occlusion.
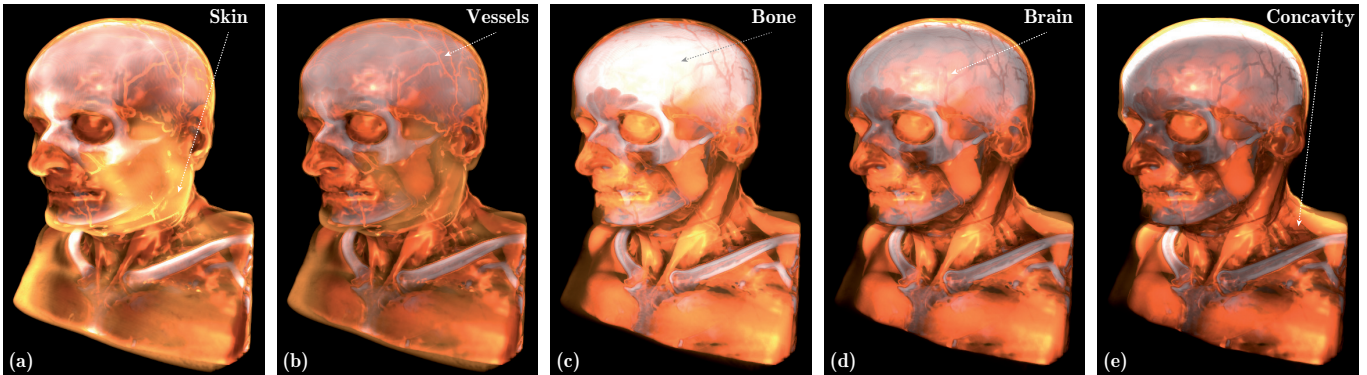

Fig. 7. Visualization of the Manix data set with ambient scattering and varying anisotropy values. The first light source is located in front of the data set and the second one is located behind. The anisotropy parameter of the HG phase function varies with (a) $g = -0.8$, (b) $g = -0.6$, (c) $g = 0.0$ (d) $g = 0.6$, and (e) $g = 0.8$. All other parameters remain constant.

ward. The same observation applies to forward scattering; however, for thin media, light is mainly transported in the direction of propagation. Overall, we can see that light transport is similar to isotropic scattering for optically thick media, independent of the anisotropy, which is the basis of the diffusion approximation [45]. However, for thin media, light transport strongly depends on the phase function. At **http://go.visus.uni-stuttgart.de/avs**, we provide preintegration tables for $\Lambda \in [0.1, 0.9]$ together with a small piece of C++ code and a short documentation that explains how the data can be loaded and evaluated.

For features like subcutaneous tissue, subsurface scattering can create a natural appearance. Figure 6 shows volume renderings of the Visible Male data set with different optical models. In Figure 6(a), single scattering with an isotropic phase function creates hard directional shadows. Alternatively, ambient occlusion creates soft local shadows in Figure 6(b). However, in both cases, the absence of multiple interactions of light leads to a low variation of luminance. In Figure 6(c), we utilize the same parameters as in Figure 6(a) for our approach. With an ambient radius of $r = 14$ voxels, the visualization appears more vivid due to subsurface scattering in the different subcutaneous structures. Our technique is combinable with gradient-based specular highlights for isosurface-like structures as shown in Figure 6(d). Furthermore, it is possible to add ambient light to illuminate dark regions. In Figure 6(e), we superimpose a small contribution of ambient occlusion to obtain a well-lit visualization.

Our method is capable of simulating a large range of anisotropy values. In Figure 7, the soft tissue and the skeleton of the Manix data set are visualized with ambient scattering. One light source is located in front of the data set, close to the observer, and one light source is located behind the data set. The lighting conditions, transfer function, and camera remain constant for this sequence of results. In Figure 7(a), strong backward scattering ($g = -0.8$) induces reflection of light back to the observer on the outmost thin layer, which emphasizes surface-like structures on the skin level. With an anisotropy value of $g = -0.6$, more light passes this thin layer and illuminates mainly the blood ves-

sels in Figure 7(b). For isotropic scattering, in Figure 7(c), light is scattered strongly in the bone structures of the skull and in the muscle tissue. By further increasing anisotropy to $g = 0.6$ in Figure 7(d), more light can pass the skull and illuminate the outer layers of the brain. In Figure 7(e), strong forward scattering ($g = 0.8$) creates a halo inside the skull from the light source behind the data set and illuminates concavities due to increased translucency.

Hard shadows can be misleading in the presence of fibrous structures. In Figure 8, the skeleton of the Manix data set is visualized with forward scattering ($g = 0.8$), where the light source is located behind the data set. In Figure 8(a), single scattering casts hard shadows of the vessels on the inside of the skull creating an illusion of line-like features. This effect becomes more apparent when the light source is moved to the left, which causes different shadows in Figure 8(b). In Figure 8(c), we keep the same lighting conditions, but we employ ambient scattering with a small radius. The shadows of the vessels appear softer, which can already help distinguish the structures. By increasing the radius in Figures 8(d) and 8(e), only the large features cast soft shadows and the dark regions of the backbone are illuminated due to increased translucency, removing misleading visual patterns.

In Figure 9, the Mecanix data set is illuminated from the left with strong forward scattering ($g = 0.8$). In Figure 9(a), single scattering is unable to transport light, since repeated in-scattering is not accounted for. In contrast, ambient scattering allows one to control the amount of translucency with the radius. In Figure 9(b), we employ ambient scattering with $r = 6$ voxels, which illuminates the far side of the torso and creates softer shadows. By increasing the radius to $r = 14$ voxels in Figure 9(c), subtle details like the navel or the abdominal muscles become visible as a consequence of multiple scattering.

Since our preintegration method does not depend on a specific data set, ambient scattering is well-suited for visualizing time-dependent data. Figure 10 shows one time step of a transient temperature field inside a closed room from a CFD simulation. On the ceiling, a cooling plate is installed with a fixed temperature of 278K (blue), while on
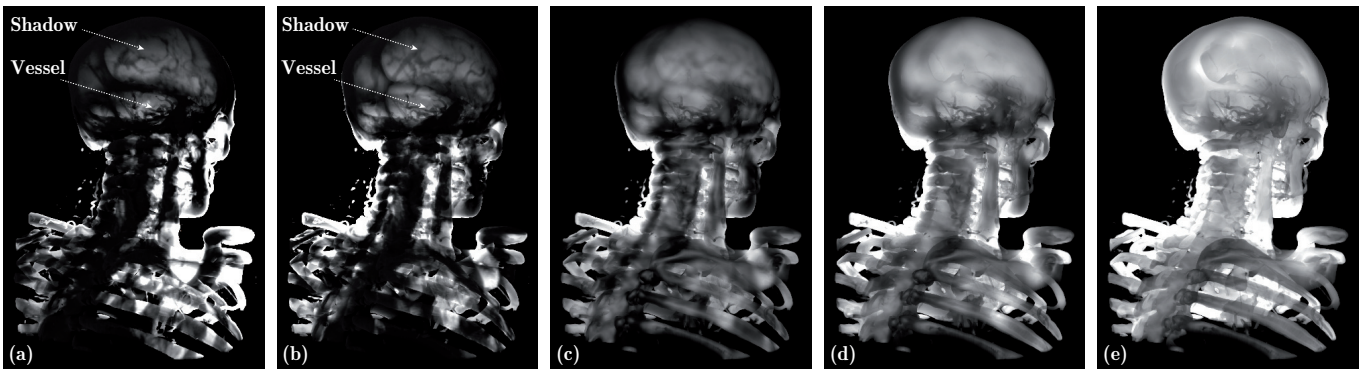
Fig. 8. Visualization of the skeleton of the Manix data set with strong forward scattering $g = 0.8$. The light source is located behind the data set. (a) Single scattering causes misleading shadows of the blood vessels on the skull. (b) The light source is moved slightly to the left, which leads to different shadows. The same lighting conditions as in (b), but with ambient scattering and a radius of (c) $r = 2$, (d) $r = 6$, and (e) $r = 14$ voxels.
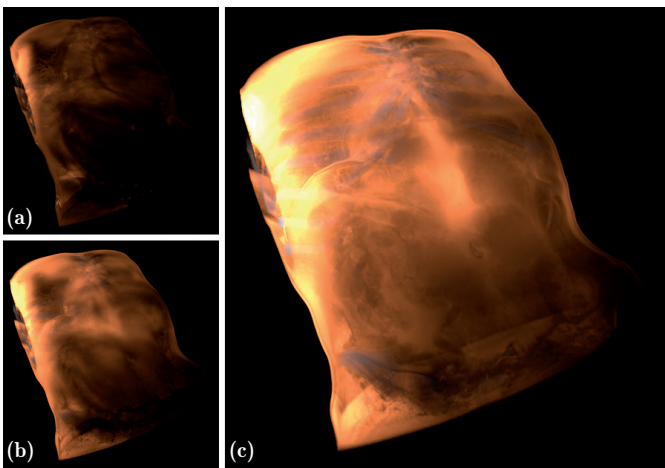

Fig. 9. The Mecanix data set with strong forward scattering $g = 0.8$. (a) Single scattering leads to strong darkening and hard shadows. Ambient scattering with a radius of (b) $r = 6$ and (c) $r = 14$ voxels illuminates subtle details by means of translucency and soft shadows.
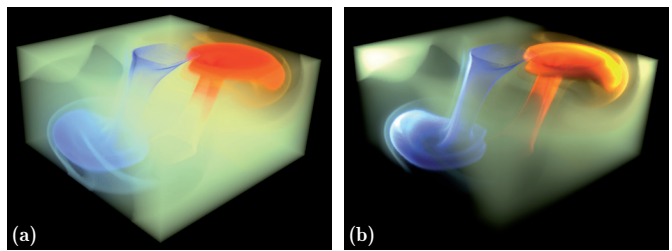

Fig. 10. Visualization of one time step of the time-dependent Buoyancy Flow data set. Cool air (blue) drops from the cooled ceiling, while hot air (red) raises from the heated floor. Due to convection, the flow becomes time-dependent. (a) Emission–absorption model. (b) Ambient scattering $g = 0.5$ with two light sources and a radius of $r = 10$ voxels.

the floor, a heating plate is located with a fixed temperature of 348K (red). Compared to the emission–absorption model in Figure 10(a), our technique provides more visual cues to determine the shape of the volumetric features in Figure 10(b) and still achieves high frame rates. Due to the short delay times for recomputing the illumination in each time step, the sequence can be visualized at interactive speed. In the supplemental video, we show the animation using both optical models.

In Figure 11, a time-dependent CFD data set [37] of a flat-plate turbulent boundary layer is visualized by means of the λ_2 vortex criterion, a derived scalar field widely used for flow visualization of vortices. Flow direction is from right to left in this figure. The data set is illuminated with (a) ambient occlusion, (b) single scattering with $g = 0.5$, and (c) ambient scattering with $g = 0.5$ and radius $r = 14$ voxels. Ambient occlusion and single scattering provide only limited insight, both with respect to the spatial organization of the complex vortical structures and the distribution of the λ_2 field within individual vortices. It is a drawback common to the traditional visualization of λ_2 using isosurfaces that these techniques cannot visualize both the outer boundary of vortices and their inner structure at the same time. The visualization based on ambient scattering, in contrast, provides at the same time the overall distribution of the λ_2 field and its spatial structure. Thus, it not only allows the researcher to judge the qualitative behavior of turbulent flows, but it provides at the same time a quantitative representation of the vortical dynamics. The reader is referred to the supplemental video for an animation of the entire sequence.

In the supplemental results, we also provide a comparison of ambient scattering with path tracing to qualitatively evaluate the visual differences of our approach with respect to full global illumination.

In Table 1, we summarize performance measurements of all results. Changing the transfer function is the most expensive kind of interaction, since the entire illumination needs to be recomputed. For extinction-based ambient occlusion and our approach, the SAT needs to be updated. According to Schlegel et al. [43], at least three evaluations of the SAT are required to compute the occlusion factor, whereas our approach requires only one evaluation to compute the ambient extinction volume. However, our method further needs to update the radiance cache when the transfer function is changed, which consumes part of the gained advantage. Nonetheless, the combined computation time remains on par with ambient occlusion, but with scattering and soft shadows already included. Changing the transfer function with single scattering is up to an order of magnitude slower compared to our approach, since small steps are required to recompute shadows without artifacts. Changing the anisotropy of the phase function affects only our method and requires a recomputation of the radiance cache. However, according to Table 1, this step is interactive for all tested results. With increasing radii, our approach benefits from larger step sizes, which can be seen in Figures 8 and 9. Moving the light sources relative to the data set is computationally equivalent with a change of the phase function with our method and requires a recomputation of the radiance cache. In contrast, single scattering needs to update the transmittance cache, which is again very expensive.

We measured rendering performance with updated caches by rotating the camera around the data set and by averaging over 720 frames. Rendering with emission–absorption achieves the highest frame rates. However, the high performance comes at the cost of few visual cues for depth and size. Single scattering and ambient occlusion require texture look-ups for transmittance and occlusion, respectively, but they
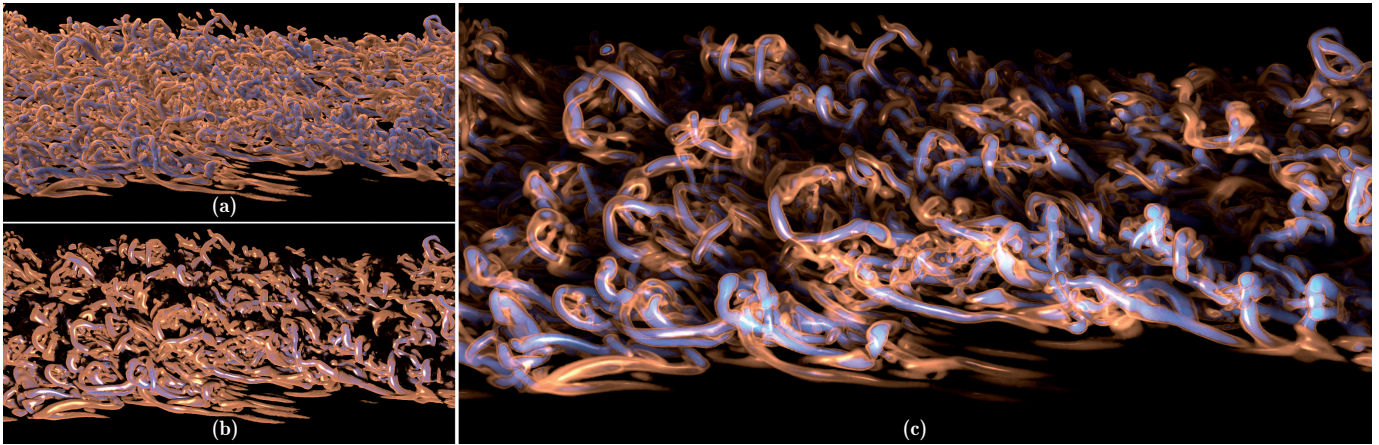
Fig. 11. Visualization of the time-dependent λ_2 field of the Boundary Layer data set with different optical models. (a) Ambient occlusion with a radius of $r = 14$ voxels. (b) Illumination with single scattering ($g = 0.5$) and additional specular highlights. (c) Ambient scattering ($g = 0.5$) and a radius of $r = 14$ voxels together with specular highlights.

Table 1. Performance measurements for the data sets of this paper. Rendering performance was obtained with a viewport of 512^2 pixels and early ray termination with a threshold of $\alpha = 0.98$ for all measurements. The computation times for changing the transfer function (TF), phase function (PF), and light sources (LS) are not included in the rendering performance. The different optical models are color-coded as follows: emission–absorption, ambient occlusion, single scattering, ambient scattering.

Data set	Volume size	Figures	TF change [ms]	PF change [ms]	LS change [ms]	Rendering [fps]
Supernova	$432 \times 432 \times 432$	1a/b/c	$-/532/490$	$-/-/34$	$-/-/34$	64/27/20
Vis. Human	$202 \times 256 \times 259$	6a/b/c/d/e	1020/126/117/117/158	$-/-/16/16/16$	$1020/-/16/16/16$	110/105/60/53/50
Manix (Tiss.)	$256 \times 256 \times 276$	7a–e	182	11	11	38
Manix (Skel.)	$256 \times 256 \times 276$	8a–b/c/d/e	1190/231/192/180	$-/36/16/10$	1190/36/16/10	60/35/35/35
Mecanix	$256 \times 256 \times 302$	9a/b/c	1372/206/192	$-/34/20$	1372/34/20	73/48/48
Buoyancy Flow	$181 \times 91 \times 181$	10a/b	$-/85$	$-/5$	$-/5$	180/63
Boundary Layer	$538 \times 54 \times 261$	11a/b/c	128/780/112	$-/-/8$	$-/780/8$	105/87/69

provide either hard and directional or soft and local shadows, both with similar rendering speed. Ambient scattering requires additional memory access for the look-up in the preintegration table, which accounts for anisotropic scattering, translucency, as well as directional and soft shadows. The extra texture access can be avoided in empty regions of the volume ($\sigma_s = 0$), since no contribution due to scattering can be expected. Although our approach is not as fast as ambient occlusion or single scattering, real-time frame rates can still be achieved. Even a per-frame recomputation of the soft shadows is possible with interactive performance, which we demonstrate at the end of the supplemental video. In general, rendering performance decreases about linearly with the number of light sources because the inner loop of Algorithm 1 is executed once for each light. With the data set sizes used in our results, we achieved interactive frame rates with up to 3 light sources. Our method is limited to point and directional light sources due to the distribution step in the rendering algorithm.

7 CONCLUSION AND FUTURE WORK

We have presented ambient volume scattering as an efficient high-quality illumination technique, including subsurface scattering, translucency, and soft shadows. In particular, our method can simulate anisotropic scattering in the ambient region of each point in the volume. In this spirit, ambient scattering complements ambient occlusion with indirect lighting based on physical light transport. Depending on the anisotropy, different features of the volume can be highlighted without changing their opacity. Compared to gradient-based shading or shadow-based ray casting, ambient scattering is more robust against sampling artifacts. It is well-suited to explore data sets, since all relevant parameters can be changed interactively, which also facilitates illumination of time-dependent data. Our algorithm can be implemented with well-known basic techniques like path tracing and ray casting.

In a practical scenario, all preintegration tables would be delivered with the system before initial operation because preintegration is independent of the data set. Hence, an operator would not need to perform a lengthy initialization process and since there is no other preprocessing step required, volume rendering is available instantly, which is crucial for time-critical applications like medical visualization.

Our method is limited by the fact that far-range illumination effects are not taken into account; hence, a solution of global illumination is not supported. This limits the use of ambient scattering for surface-based illumination, where much empty space is in between the patches and where interreflections are not negligible over far distances. For volume visualization, however, the use of our variable finite region offers high performance for data exploration and accounts for the most relevant contributions of scattered light.

As future work, we want to evaluate the perceptual cues of our approach compared to more advanced illumination techniques by means of a user study, similar to Lindemann and Ropinski [28]. Furthermore, we plan to accelerate ambient scattering with common methods from DVR like multiresolution rendering, which can be readily applied to our algorithm. Apart from rendering speed, further interaction concepts with ambient scattering could be investigated, for example, the interactive manipulation of the preintegration table by means of a new transfer function for data-driven illumination.

REFERENCES

[1] M. Ament, D. Weiskopf, and H. Carr. Direct interval volume visualization. *IEEE Transactions on Visualization and Computer Graphics*, 16(6):1505–1514, 2010.

[2] 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.

[3] S. Bruckner and M. E. Gröller. Enhancing depth-perception with flexible volumetric halos. *IEEE Transactions on Visualization and Computer Graphics*, 13(6):1344–1351, 2007.

[4] E. Cerezo, F. Pérez, X. Pueyo, F. J. Seron, and F. X. Sillion. A survey on participating media rendering techniques. *The Visual Computer*, 21(5):303–328, 2005.

[5] S. Chandrasekhar. *Radiative Transfer*. Dover, 1960.

[6] F. C. Crow. Summed-area tables for texture mapping. *SIGGRAPH Computer Graphics*, 18(3):207–212, 1984.

[7] B. Csébfalvi and L. Szirmay-Kalos. Monte Carlo volume rendering. In *Proc. of IEEE Visualization*, pages 449–456, 2003.

[8] P. Desgranges, K. Engel, and G. Paladini. Gradient-free shading: A new method for realistic interactive volume rendering. In *Proc. of the Workshop on Vision, Modeling, and Visualization*, pages 209–216, 2005.

[9] J. Díaz, P.-P. Vázquez, I. Navazo, and F. Duguet. Real-time ambient occlusion and halos with summed area tables. *Computers & Graphics*, 34(4):337–350, 2010.

[10] K. Engel, M. Kraus, and T. Ertl. High-quality pre-integrated volume rendering using hardware-accelerated pixel shading. In *Proc. of the ACM SIGGRAPH/Eurographics Workshop on Graphics Hardware*, pages 9–16, 2001.

[11] A. Guetat, A. Ancel, S. Marchesin, and J.-M. Dischler. Pre-integrated volume rendering with non-linear gradient interpolation. *IEEE Transactions on Visualization and Computer Graphics*, 16(6):1487–1494, 2010.

[12] 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.

[13] J. Hensley, T. Scheuermann, G. Coombe, M. Singh, and A. Lastra. Fast summed-area table generation and its applications. *Computer Graphics Forum*, 24(3):547–555, 2005.

[14] L. Henyey and J. Greenstein. Diffuse radiation in the galaxy. *Astrophysical Journal*, 93:70–83, 1941.

[15] F. Hernell, P. Ljung, and A. Ynnerman. Interactive global light propagation in direct volume rendering using local piecewise integration. In *Proc. of the IEEE/EG International Symposium on Volume and Point-Based Graphics*, pages 105–112, 2008.

[16] F. Hernell, P. Ljung, and A. Ynnerman. Local ambient occlusion in direct volume rendering. *IEEE Transactions on Visualization and Computer Graphics*, 16(4):548–559, 2010.

[17] H. W. Jensen and P. H. Christensen. Efficient simulation of light transport in scenes with participating media using photon maps. In *Proc. of ACM SIGGRAPH*, pages 311–320, 1998.

[18] H. W. Jensen, S. R. Marschner, M. Levoy, and P. Hanrahan. A practical model for subsurface light transport. In *Proc. of ACM SIGGRAPH*, pages 511–518, 2001.

[19] D. Jönsson, J. Kronander, T. Ropinski, and A. Ynnerman. Historygrams: Enabling interactive global illumination in direct volume rendering using photon mapping. *IEEE Transactions on Visualization and Computer Graphics*, 18:2364–2371, 2012.

[20] D. Jönsson, E. Sundén, A. Ynnerman, and T. Ropinski. Interactive volume rendering with volumetric illumination. In *Eurographics STAR program*, pages 53–74, 2012.

[21] J. T. Kajiya. The rendering equation. *SIGGRAPH Computer Graphics*, 20(4):143–150, 1986.

[22] J. Kniss, S. Premože, C. Hansen, and D. Ebert. Interactive translucent volume rendering and procedural modeling. In *Proc. of IEEE Visualization*, pages 109–116, 2002.

[23] J. Kniss, S. Premože, C. Hansen, P. Shirley, and A. McPherson. A model for volume lighting and modeling. *IEEE Transactions on Visualization and Computer Graphics*, 9(2):150–162, 2003.

[24] A. Knoll, Y. Hijazi, R. Westerteiger, M. Schott, C. Hansen, and H. Hagen. Volume ray casting with peak finding and differential sampling. *IEEE Transactions on Visualization and Computer Graphics*, 15(6):1571–1578, 2009.

[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 Transactions on Visualization and Computer Graphics*, 18(3):447–462, 2012.

[27] M. Langer and H. Bülthoff. Depth discrimination from shading under diffuse lighting. *Perception*, 29(6):649–660, 2000.

[28] F. Lindemann and T. Ropinski. About the influence of illumination models on image comprehension in direct volume rendering. *IEEE Transactions on Visualization and Computer Graphics*, 17(12):1922–1931, 2011.

[29] E. B. Lum and K.-L. Ma. Lighting transfer functions using gradient aligned sampling. In *Proc. of IEEE Visualization*, pages 289–296, 2004.

[30] E. B. Lum, B. Wilson, and K.-L. Ma. High-quality lighting and efficient pre-integration for volume rendering. In *Proc. of the Joint Eurographics - IEEE TCVG Conference on Visualization*, pages 25–34, 2004.

[31] M. Magnor, K. Hildebrand, A. Lintu, and A. Hanson. Reflection nebula visualization. In *Proc. of IEEE Visualization*, pages 255–262, 2005.

[32] N. Max. Optical models for direct volume rendering. *IEEE Transactions on Visualization and Computer Graphics*, 1(2):99–108, 1995.

[33] J. T. Moon, B. Walter, and S. R. Marschner. Rendering discrete random media using precomputed scattering solutions. In *Proc. of the Eurographics Conference on Rendering Techniques*, pages 231–242, 2007.

[34] M. Pharr and G. Humphreys. *Physically Based Rendering*. Elsevier, 2004.

[35] B. T. Phong. Illumination for computer generated pictures. *Communications of the ACM*, 18(6):311–317, 1975.

[36] C. Rezk-Salama. GPU-based Monte-Carlo volume raycasting. In *Proc. of the Pacific Conference on Computer Graphics and Applications*, pages 411–414, 2007.

[37] U. Rist. Visualization and tracking of vortices and shear layers in the late stages of boundary-layer laminar-turbulent transition. In *50th AIAA Aerospace Sciences Meeting*, 2012.

[38] 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.

[39] T. Ropinski, J. Meyer-Spradow, S. Diepenbrock, J. Mensmann, and K. Hinrichs. Interactive volume rendering with dynamic ambient occlusion and color bleeding. *Computer Graphics Forum*, 27(2):567–576, 2008.

[40] S. Röttger, M. Kraus, and T. Ertl. Hardware-accelerated volume and isosurface rendering based on cell-projection. In *Proc. of IEEE Visualization*, pages 109–116, 2000.

[41] M. Ruiz, I. Boada, I. Viola, S. Bruckner, M. Feixas, and M. Sbert. Obscurance-based volume rendering framework. In *Proc. of the IEEE/EG International Symposium on Volume and Point-Based Graphics*, pages 113–120, 2008.

[42] H. E. Rushmeier. *Realistic image synthesis for scenes with radiatively participating media*. PhD thesis, Cornell University, 1988.

[43] P. Schlegel, M. Makhinya, and R. Pajarola. Extinction-based shading and illumination in GPU volume ray-casting. *IEEE Transactions on Visualization and Computer Graphics*, 17(12):1795–1802, 2011.

[44] M. Schott, V. Pegoraro, C. Hansen, K. Boulanger, and K. Bouatouch. A directional occlusion shading model for interactive direct volume rendering. *Computer Graphics Forum*, 28(3):855–862, 2009.

[45] J. Stam. Multiple scattering as a diffusion process. In *Proc. of the Eurographics Workshop on Rendering*, pages 51–58, 1995.

[46] A. J. Stewart. Vicinity shading for enhanced perception of volumetric data. In *Proc. of IEEE Visualization*, pages 355–362, 2003.

[47] E. Sundén, A. Ynnerman, and T. Ropinski. Image plane sweep volume illumination. *IEEE Transactions on Visualization and Computer Graphics*, 17(12):2125–2134, 2011.

[48] C. Wyman, S. Parker, P. Shirley, and C. Hansen. Interactive display of isosurfaces with global illumination. *IEEE Transactions on Visualization and Computer Graphics*, 12(2):186–196, 2006.

[49] C. Zhang and R. Crawfis. Shadows and soft shadows with participating media using splatting. *IEEE Transactions on Visualization and Computer Graphics*, 9(2):139–149, 2003.

[50] 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.