

A Multidirectional Occlusion Shading Model for Direct Volume Rendering

Veronika Šoltészová¹ Daniel Patel² Stefan Bruckner³ Ivan Viola¹

¹University of Bergen, Norway

²Christian Michelsen Research, Norway

³Simon Fraser University, Canada

Abstract

In this paper, we present a novel technique which simulates directional light scattering for more realistic interactive visualization of volume data. Our method extends the recent directional occlusion shading model by enabling light source positioning with practically no performance penalty. Light transport is approximated using a tilted cone-shaped function which leaves elliptic footprints in the opacity buffer during slice-based volume rendering. We perform an incremental blurring operation on the opacity buffer for each slice in front-to-back order. This buffer is then used to define the degree of occlusion for the subsequent slice. Our method is capable of generating high-quality soft shadowing effects, allows interactive modification of all illumination and rendering parameters, and requires no pre-computation.

Categories and Subject Descriptors (according to ACM CCS):

I.3.7 [Computer Graphics]: Three-Dimensional Graphics and Realism —Color, shading, shadowing, and texture

1. Introduction

Local illumination models, such as the Phong model, are suited for conveying shape cues for well-defined structures in volume data. However, they are generally not suitable for visualization when the main goal is to emphasize three-dimensional structural arrangements. In such a scenario, it is important to convey information about relative positions and distances between individual features. The human visual system is adapted to inferring three-dimensional information from illumination. Soft shadows, in particular, are effective monocular depth cues. Not only do they provide occlusion information, but the size and shape of the penumbra can be used to judge distances. Global illumination models provide these cues at high computational costs, especially for volume rendering. Visualization research has therefore focused on the development of new global illumination approximations for volume data that limit the complexity and allow for real-time image synthesis. For this purpose, precomputation or parameter constraint strategies are frequently employed. Both suffer from limited flexibility which can be problematic when interactive exploration is required. The directional oc-

clusion shading model introduced by Schott et al. [SPH*09] is a forward scattering approximation based on a conical phase function. While the method is capable of generating realistic illumination at interactive frame rates, it requires that the view and the light directions have to coincide. In this paper, we introduce a multidirectional occlusion model, which removes this constraint.

The importance of illumination in 3D object perception has been well-studied [BBC83,KMK94,BLK00]. To find out how to best improve volume rendering, we have been conducting studies with medical illustrators. During our demonstrations of state-of-the-art visualization techniques to experienced medical illustrators, their first critique point was the positioning of the light in the scene and the choice of non-standard colors. While visualization researchers often carelessly define the light vector parallel to the view vector, this is considered a novice mistake in the domain of illustration. The resulting image is *flat*, akin to photos taken with built-in front flash. To give depth to an image, as a rule, medical illustrators use illumination from the top left. To further op-

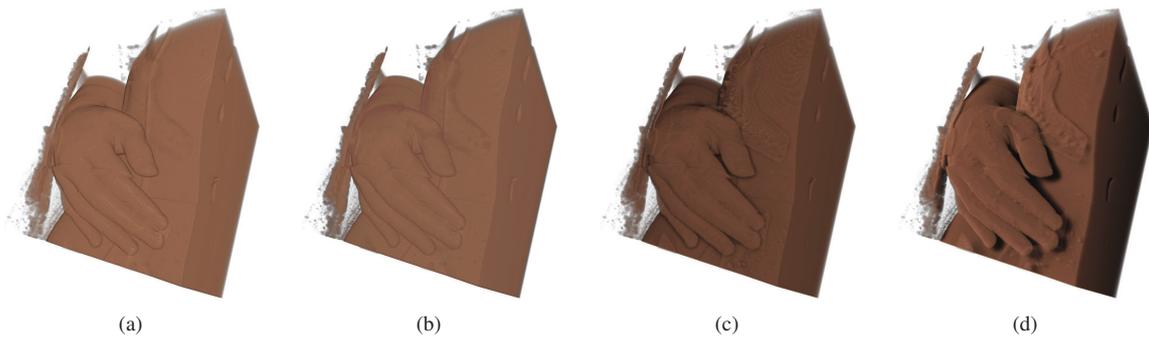


Figure 1: Visualizations of a human hand using raycasting (a), and sliced-based volume rendering (b), both using the Phong illumination. Directional occlusion shading model with a headlamp illumination setup (c) and illumination from the top left (d). Illumination with the top-left light position causes that the fingers cast soft shadows on the body and evoke strong depth-perception cues.

timize the appearance of the depicted structures, manual fine tuning is required.

The directional occlusion shading model for interactive direct volume rendering takes the advantage of a headlight setup for performance reasons: by placing the light source at the eye position, the samples required for illumination can be reused for compositing, allowing the method to perform both operations in a single pass for a view-aligned slice through the volume. Our approach uses elliptic occlusion footprints computed from the light position, instead of the symmetric spherical footprints which were used in the original paper. We achieve the same performance with the additional possibility to position the light source anywhere within the hemisphere defined by the view vector. An example of the significant improvement of depth perception compared to the previous method is shown in Figure 1. A visualization of a human hand rendered with different techniques is compared to the headlight and top-left shading. Both a professional illustrator and a user study confirmed our subjective assessment which favored the rendering result 1(d).

The remainder of this paper is structured as follows: In Section 2, we review related work. Our multidirectional occlusion model is derived in Section 3. Section 4 provides additional implementation details. Results are presented and discussed in Section 5. Conclusions are drawn in Section 6.

2. Related Work

Levoy [Lev87] proposed the use of gradients in volume rendering for evaluating a surface-based local illumination model. While this common approach is effective in accentuating material boundaries, it suffers from noise. In particular, gradient-based shading fails to provide useful results in nearly homogenous regions. Illumination models which exploit the volumetric nature of the data can therefore pro-

vide additional cues. Max [Max95] gives a comprehensive overview of different optical models for volume rendering.

Yagel et al. [YKZ91] employed recursive ray tracing which allows for effects such as specular reflection and shadows. Behrens and Ratering [BR98] added shadows to texture-based volume rendering by using an additional shadow volume. The model presented by Kniss et al. [KKH02, KPH*03] captures volumetric light attenuation effects including volumetric shadows, phase functions, forward scattering, and chromatic attenuation using half-angle slicing. Hadwiger et al. [HKSB06] presented a GPU-accelerated algorithm for computing deep shadow maps for volume rendering. Rezk-Salama [RS07] proposed a semi-interactive approach for GPU-based Monte Carlo volume raytracing.

Ambient occlusion as described by Zhukov et al. [ZIK98] inverts the principle of light-exposure of a point in space to obscurity by its close environment. Dachsbacher et al. [DSDD07] refer to obscurity as antiradiance. They treat visibility implicitly while propagating antiradiance as an additional quantity. The advantage of these approaches is that they are view-independent: for fixed geometry, occlusion information only has to be computed once and can then be applied efficiently during rendering, for example using texture mapping. Several fast techniques which utilize this concept have been presented [Bun05, SA07]. Knecht [Kne07] and Méndez-Feliu [MS09] provide comprehensive overviews of rendering techniques based on ambient occlusion and obscurities.

In the context of volume visualization, the radiance at a point is determined by shooting rays in all directions from the point and averaging its degree of occlusion by other parts of the volume. The result is an approximation of global diffuse illumination. It produces soft shadowing effects which give a good indication of spatial relationships. However, the

opacity at any point is determined by the transfer function. Ambient occlusion therefore requires an expensive computation step every time the transfer function is modified. Stewart [Ste03] introduced vicinity shading, a variation of ambient occlusion to enhance perception of volume data by darkening depressions and crevices. To reduce evaluation costs, occlusion calculations are reused. The approach of Ropinski et al. [RMSD*08] relied on local histogram clustering to precompute occlusion information for all possible transfer function settings. However, high frequency data, in particular the presence of noise, reduces the effectiveness of their clustering approach and can lead to artifacts. Additionally, their precomputation process is very time and memory consuming. Hernell et al. [HYL07] used a local approximation of ambient occlusion in volumes to limit computation times. In subsequent work [HLY08, HLY09], they utilized local piecewise integration to approximate global light propagation. This approach still requires ambience data for each voxel to be recomputed when changing the transfer function, but their method is able to run interactively by limiting the number of rays shot for evaluating the ambience and by subsampling the rays using adaptive compression. In recent work, Ropinski et al. [RDR10] described a volumetric lighting model which simulates scattering and shadowing. They use slice-based volume rendering from the view of the light source to calculate a light volume and raycasting to render the final image.

View-dependent approaches do not require extensive pre-computation and therefore allow fully interactive transfer function modification. This is frequently achieved by limiting light evaluation from spherical local neighborhoods to conical neighborhoods. Desgranges et al. [DEP05] use incremental blurring to achieve shading effects without the use of a gradient. The approach by Bruckner and Gröller [BG07] employed non-photorealistic shadowing and emission effects for the purpose of illustration. Finally, as stated in the previous section, our method is an extension of the model by Schott [SPH*09].

3. Multidirectional Occlusion Shading

The Directional Occlusion Shading model by Mathias Schott et al. (MS-DOS) [SPH*09] describes an approximation of light scattering in particles of a volume. This simple method generates soft a shadow effect and hence provides important shape and depth-perception cues. Although the approximation of the light transfer delivers slightly different results compared to reference images from a raytracer, it provides visually compelling shading effects at interactive frame-rates and with no precomputation. However, the light transfer approximation in the MS-DOS model constrains the light direction to the viewing direction. In this section, we derive an approximation which does not limit the light to this fixed direction.

3.1. Light Transport Equation

The directional occlusion shading model approximates transport of light energy L in a medium. Every point in the environment receives a portion of energy, i.e., radiance composed by background radiance L_b and medium radiance L_m . The medium radiance consists of the emitted radiance L_e and in-scattered radiance L_i . The emitted radiance at a point \mathbf{x} depends only on the local environment of \mathbf{x} . Unlike L_e , the in-scattered radiance L_i integrates over global features:

$$L_i(\mathbf{x}, \omega) = \int_{4\pi} L(\mathbf{x}, \omega_i) \Phi(\omega, \omega_i) d\omega_i \quad (1)$$

where $\Phi(\omega, \omega_i)$ denotes the phase function for two light-ray directions ω and ω_i . L_i quantifies the total radiance incident to point \mathbf{x} from all directions ω_i . From Equation 1, it can be seen that L_i requires an expensive recursive evaluation. The MS-DOS shading model and our model (multidirectional OS) simplify the evaluation which considerably reduces the computational costs.

We assume that the medium emits light only in directions within a specific cone. The phase function from Equation 1 can be therefore replaced by a simple cone-shaped phase function $\Phi_{\theta, \alpha}(\omega, \omega_i)$ where θ is the aperture angle and α the tilt angle of the cone. A schematic illustration of this scenario is depicted in Figure 2. A particle at a point \mathbf{x} scatters light which is received by particles inside the cone. The in-scattering term L_i is conceptually related to the fractional visibility which is equivalent to the opacity and cumulates information about ambient occlusion.

Like the original model, we use a slice-based volume renderer with an additional opacity buffer. Slices are composited in the front-to-back order and the opacity buffer is incrementally filtered and used to determine the accumulated opacity for the next slice as shown in Figure 3. MS-DOS

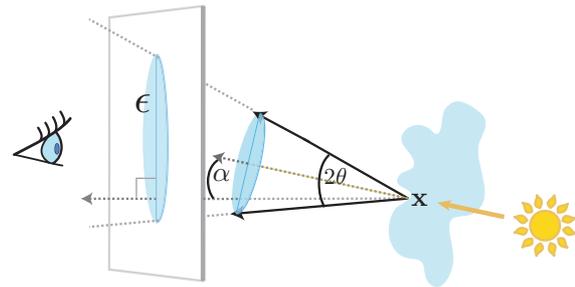


Figure 2: Conical phase function setup: a selected point in space \mathbf{x} scatters light which we approximate by a tilted cone ($\alpha =$ tilt, $\theta =$ aperture). The axis of the cone is parallel to the light direction. The projection of the light energy leaves an elliptical footprint ϵ on a selected viewing plane.

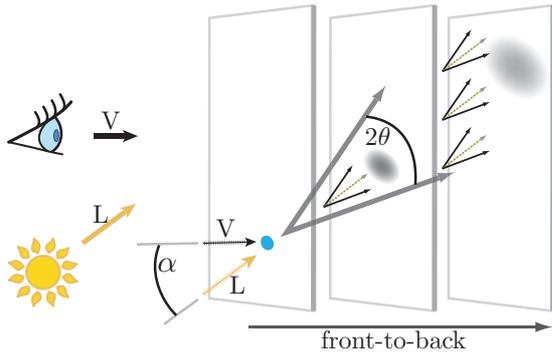


Figure 3: Incremental blurring of the opacity buffer. We use a view-aligned slice stack composited in the front-to-back order.

operates on view-aligned slices and assumes that the direction of the light source is aligned to the viewing direction. As a consequence, the opacity buffer can be convolved with a symmetrical disc-shaped Gaussian kernel. To enable interaction with the light source, we change the symmetrical disc-shaped kernel to an elliptical kernel. The ellipse ϵ is defined by the intersection of a tilted cone which represents the phase function $\Phi_{\theta, \alpha}(\omega, \omega_i)$ and the slice plane. The cone-shaped phase function is tilted by an angle α which is limited to $[0, \frac{\pi}{2} - \theta)$. This restricts the cone-section from degenerating into hyperbolas or parabolas. Figure 4 describes this geometrical situation.

3.2. Analysis of the Geometrical Model

This section describes the analytical computation of the elliptical kernel, namely, the derivation of its major and minor axes $A = |A_1A_2|$ and $B = |CC'|$ from a known tilt α , a cone aperture θ and a slice distance d . According to Figure 4(a), we derive R from d , θ and α as:

$$R = d \frac{\tan \theta}{\cos \alpha} \quad (2)$$

The axis of the cone intersects the plane at the point O . When the tilt angle $\alpha = 0$, the cone section is a circle, and $a_1 = a_2 = A$. With a known R , we turn to the law of sine in the triangles $\triangle A_1V_1O$ and $\triangle OA_2V_2$. With α , θ , and R given, Equations 3 and 4 yield a_1 and a_2 :

$$\frac{a_1}{\sin(\frac{\pi}{2} - \theta)} = \frac{R}{\sin(\frac{\pi}{2} + \theta - \alpha)} \quad (3)$$

$$\frac{a_2}{\sin(\frac{\pi}{2} + \theta)} = \frac{R}{\sin(\frac{\pi}{2} - \theta - \alpha)} \quad (4)$$

$$A = \frac{a_1 + a_2}{2} \quad (5)$$

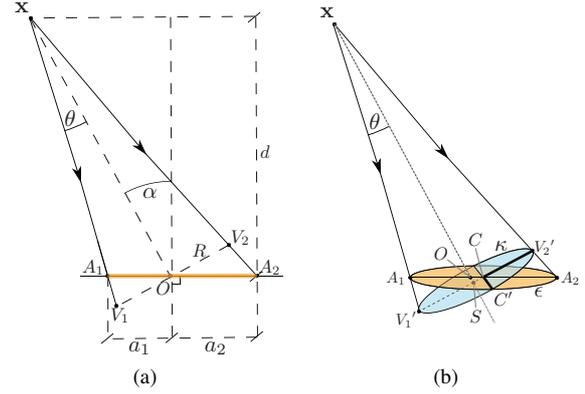


Figure 4: A geometrical description of the cone-shaped phase function: the elliptical cone-section defines a circle κ centered in S and intersecting the center C of the ellipse ϵ . A side view (a) and a 3D view (b) of the planar cone-section.

With known a_1 and a_2 , we use Equation 5 to calculate A which is the major axis of the ellipse.

The center of the ellipse ϵ is in C with $\|OC\| = \|\frac{a_2 - a_1}{2}\|$. We define a circular cone section κ which intersects the point C . Consequently, the axis of the cone intersects κ in its center S . This scenario is illustrated in Figure 4(b). The intersection line $\epsilon \cap \kappa$ is perpendicular to A_1A_2 and intersects the center C of ϵ . Consequently, $\epsilon \cap \kappa$ is collinear with the minor axis of ϵ . Figure 5 illustrates the side view of $\epsilon \cap \kappa$. In Figure 6, we focus on the triangles $\triangle XV_2'V_2$ and $\triangle SCO$, and on the circle κ . Basic analysis implies Equations 6, 7, 8, and 9. Solving them, we determine B - the minor axis of ϵ .

$$\sin \alpha = \frac{d'}{\|OC\|} \quad (6)$$

$$\tan \theta = \frac{dR}{d'} \quad (7)$$

$$R' = R + dR \quad (8)$$

$$B = \sqrt{R'^2 - \|OC\|^2 + d'^2} \quad (9)$$

3.3. Weighting Function

Light rays collinear to the cone axis hit the slice with the highest intensity. We revisit Figure 4(b): O is the point with the highest incident energy. We define a weighting function as follows:

$$W_L(x, y) = 1 - k \quad (10)$$

with k defined implicitly by:

$$\frac{(x - (1 - k)\|OC\|)^2}{A^2} + \frac{y^2}{B^2} = k^2 \quad (11)$$

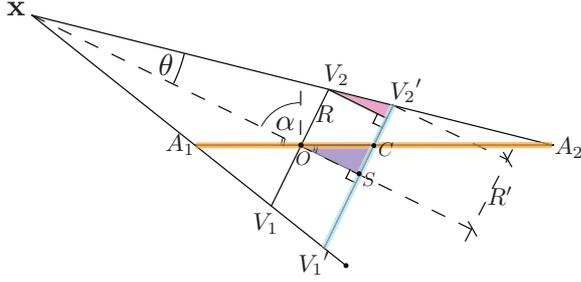


Figure 5: A detailed side view of the intersection of ellipse ε and circle κ .

Equation 11 results in a quadratic equation with two real roots from which we take the maximum. A kernel with a linear fall-off from O towards the border of the ellipse is illustrated in Figure 7(a). Additionally, we apply the Gaussian function to smooth the fall-off of the weights as illustrated in Figure 7(b).

3.4. On-the-fly Convolution

We apply an incremental convolution of the opacity buffer O_i and the elliptical kernel G_ε for each slice i . As the light direction \mathbf{L} changes, G_ε has to be aligned respectively. We project the light vector to the viewing plane which yields a 2D-vector \mathbf{L}' and rotate the kernel so that its major axis is aligned with \mathbf{L}' :

$$\frac{\vec{OC}}{\|OC\|} = \frac{\mathbf{L}'}{\|\mathbf{L}'\|} \quad (12)$$

The weight-distribution in G_ε depends only on the tilt, aperture, light direction, and slice distance. Therefore, an update is triggered only if one of these parameters changes. In practice, we render the kernel G_ε to a texture when an update is triggered. First, we uniformly scale the ellipse so that it fits into a unit-square. Second, we set-up the texture coordinates so that G_ε is aligned correctly. During the volume rendering pass, we apply inverse scaling operation to regenerate G_ε of the correct size. In Figure 8, we visualize the gecko dataset with different tilts and apertures.

Based on the incremental convolution (*), we calculate a modulation factor λ_i for each sample on the slice i which determines the visibility of the current slice:

$$\lambda_i = \frac{1}{1 + G_\varepsilon * O_{i-1}} \quad (13)$$

In addition to the opacity buffer O_i , we use a color buffer C_i for each slice. The opacity buffer for the next slice combines the opacity buffer of the previous slice with the opacity of the current slice α_i :

$$O_i = G_\varepsilon * O_{i-1} + \alpha_i \quad (14)$$

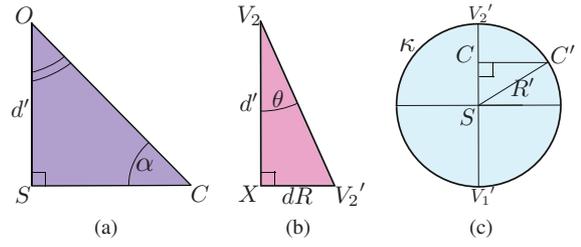


Figure 6: We introduce new literals for selected primitives: (a) the triangle ΔSCO , (b) the triangle $\Delta XV_2'V_2$ and (c) the circle κ . These primitives are defined in Figures 4 and 5 by the same color encoding. Note, that $\|CC'\| = B$ which is the minor axis of the ellipse ε .

The color contribution c_i is multiplied by λ_i . The color c_i and opacity α_i propagate to C_{i+1} using traditional alpha-blending with the over operator. Our method requires no pre-computation and performs at interactive frame-rates. Due to incremental blurring of the opacity buffer, shadows cast by highly occlusive regions fade-out smoothly with distance. Compared to the state-of-the-art model, we thereby add a movable light source with negligible performance penalty.

4. Implementation Details

Our new model was implemented as a plugin to VolumeShop [BG05] using C++ and OpenGL/GLSL. Using the ARB_draw_buffers OpenGL extension, two render targets are written for each slice: the intermediate rendered image and the occlusion buffer. The elliptical blurring kernel is stored in an additional texture which is updated whenever the light source parameters change. For all examples in the paper, we use a texture size of 128×128 . When the lighting parameters change, we recompute the footprint. The major axis of the ellipse is aligned with the projection of the light vector to the viewing plane by multiplying $GL_TEXTURE$ matrix stack by a rotation matrix. In case the ellipse grows or moves out of the texture, we apply translation and scaling to

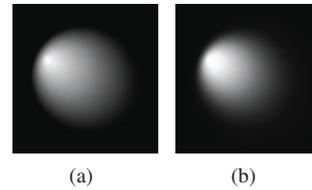


Figure 7: Elliptical kernels used for incremental blurring of the opacity buffer: with linear fall-off (a) and Gaussian fall-off of the weighting function (b).

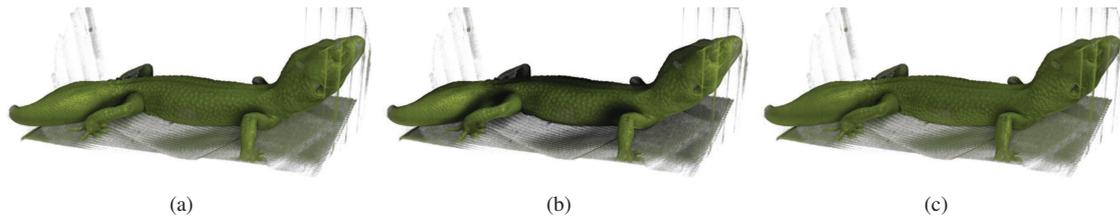


Figure 8: Visualizations of the gecko CT dataset with different setup of aperture θ and tilt angle α : $\theta = 10^\circ$ and $\alpha = 37^\circ$ (a), $\theta = 40^\circ$ and $\alpha = 37^\circ$ (b), and $\theta = 40^\circ$ and $\alpha = 5^\circ$ (c). Light is coming from the right for all images.



Figure 9: Visualizations of the carp CT dataset using the directional occlusion shading model with a headlamp illumination setup (a) and using illumination setup conventional to medical illustrations (b).

fit it into the bounding box of the texture. During rendering, the inverse transformation is applied to access the kernel at correct positions. However, massive downscaling of the coordinate system may lead to a loss of precision. Users can interactively adjust the tilt, the aperture, and the XY-rotation of the light source. This gives the user full control to set the light source arbitrarily in the hemisphere defined by the view vector. The parameters aperture, tilt, and rotation are set by sliders in the user-interface.

5. Results and Discussion

In this section, we provide case studies and comparisons to other volume rendering approaches and analyze the performance of our new method.

5.1. Case Studies

Medical illustrators generally place the light source in the top left corner to improve depth perception. Figure 9 depicts the carp CT dataset visualized under different illumination conditions. While in Figure 9(a), the image lacks depth, Figure 9(b) emphasizes the detailed structure of the skeleton through shadows. Similarly, Figure 10 shows cases where illumination leads to better perception of structures. In Figure 10(a), the hand seems to directly contact the body. In reality, there is a small gap which is visible in Figure 10(b). Similarly for Figures 10(c) and 10(d): in Figure 10(d), the eye sockets of the skull appear deeper than in Figure 10(c). We consulted a certified medical illustrator with over 25

years of professional experience who affirmed that the visualizations generated using this kind of illumination yield stronger perception cues. We presented her visualizations using different lighting settings. Her task was to choose which lighting conditions suit medical illustrations the best. She consistently preferred image such as those depicted in Figures 10(b) and 10(d). The illustrator further confirmed that interactive fine-tuning of the exact light placement is necessary in many cases, in order to avoid excessive darkening of focus objects. In volume data, regions with high gradient magnitude correspond to surface-like structures. Using the gradient magnitude to add an additional local specular component to these objects can further improve perception. Figure 11 presents a computer tomography of a human foot generated with different illumination models and varying light source positions: Figures 11(a) and 11(b) use the multidirectional OS model enhanced by specular highlights, and Figures 11(c) and 11(d) use the pure multidirectional OS model.

To gain a coarse impression on the impact of our technique on non-professionals, we also conducted a small user study on a group of 42 participants with different backgrounds. We presented them two series of result images: the human hand and the human thorax which are shown in Figures 1 and 12. Their task was to choose an image which in their opinion yields the strongest depth cues. From the series of different OS renderings of the hand, 39 participants (92.86%) favored the top-left illumination in Figure 1(d), 2 participants (4.76%) preferred the raycasting in Figure 1(a) and 1 participant (2.38%) preferred the headlamp illumination in Figure 1(c). A majority of 41 (97.62%) participants also pre-

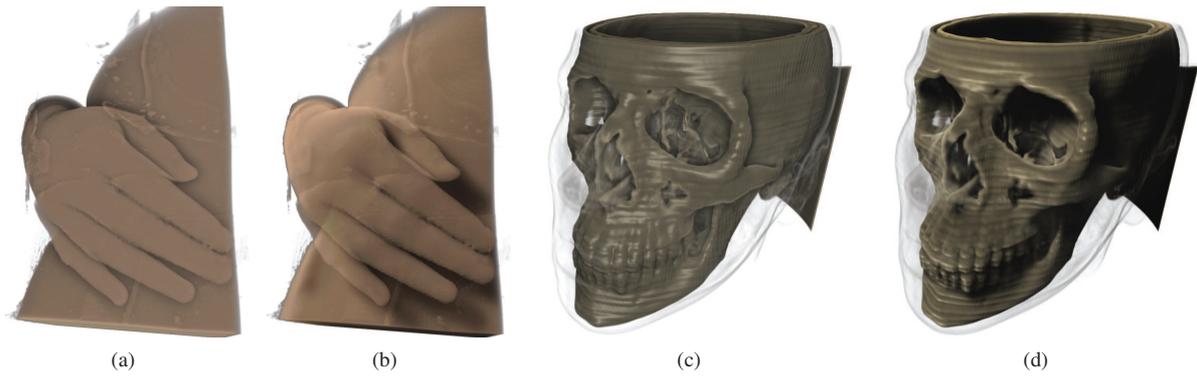


Figure 10: Visualizations of computer tomography data using the directional occlusion shading model with a headlamp illumination setup (a) and (c) using the illumination setup conventional to medical illustrations (b) and (d).



Figure 11: Visualizations of a human foot acquired by computer tomography using the directional occlusion shading model: using the Phong illumination model with the headlamp illumination setup (a) and with the top-left lighting (b). Visualizations (c) and (d) use the diffuse illumination model with the headlamp and the top-left light source setup respectively.

ferred the top-left illumination of the thorax in Figure 12(d) and only one participant (2.38%) selected the raycasted image in Figure 12(a).

Local surface-based illumination of volume data employs the gradient to substitute for the surface normal. However, gradients estimation frequently performs poor in the presence of noise which can lead to distracting artifacts. Thus, for modalities such as ultrasound, unshaded volume rendering is commonly employed. This makes the structures in the data difficult to interpret even for experienced medical professionals. The directional occlusion shading model as a gradient-free shading method can be used to improve perception. Interactive light source modification enables the user to inspect and understand the structures better. Figure 13 shows different visualizations of 3D cardiac ultrasound: 2D slices and 3D volume renderings. The clipping plane reveals the inside of the heart chambers. During examination, physicians see the ultrasound visualizations on their workstations as in Figure 13(a). We used a transfer function which shows the heart in a similar fashion. Figure 13(b) shows that gradient-based shading is not well-suited for ultrasound data. Multidirectional occlusion shading, on the other hand, reveals the structure, and interaction with light

source enables the user to better perceive the depth of the cavities.

We described a shading model which does not require precomputation and storage of additional data, unlike deep shadow maps [HKS06] or light volumes [RDR10], and which allows arbitrary light position within the hemisphere defined by the view vector. Half-angle slicing, introduced in the work of Kniss et al. [KKH02], generates shadows by using a slicing direction halfway between view and light direction. However, choosing a slicing direction which is non-parallel to the viewing directions leads to visible artifacts, especially when the light source tilt angle surpasses 60° . Figure 14 clearly demonstrates a situation when such artifacts are visible when half-angle slicing is used. In the original half-angle slicing approach, the order of the slices is reverted if the light source is located in the hemisphere opposite to the viewer. Reverting the traversal of the slice-stack is a possible extension of our approach which would not limit the light vector to the hemisphere defined by the view vector.

5.2. Performance Analysis

We tested the described method on a workstation equipped with an NVIDIA GeForce 295 GTX GPU with 1.7GB graph-

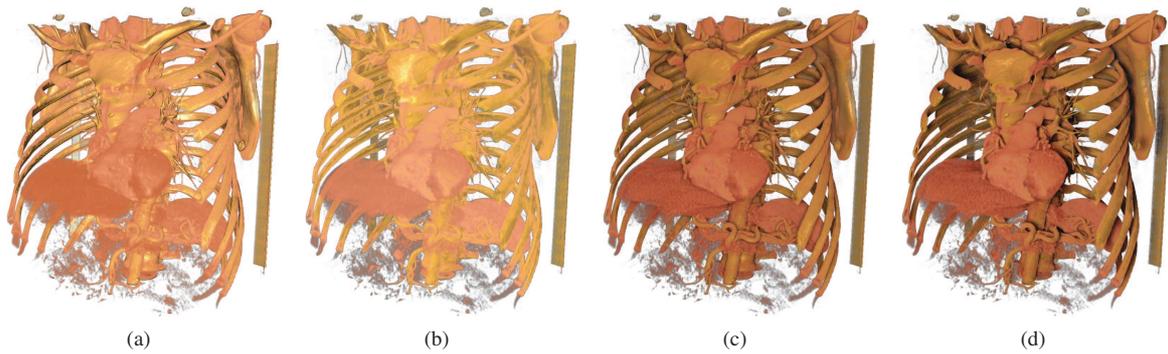


Figure 12: Visualizations of a human thorax we used for user study: using raycasting (a), sliced-based volume rendering (b), both using the Phong illumination followed by the directional occlusion shading model with the headlamp illumination setup (c) and illuminated from the top left (d).

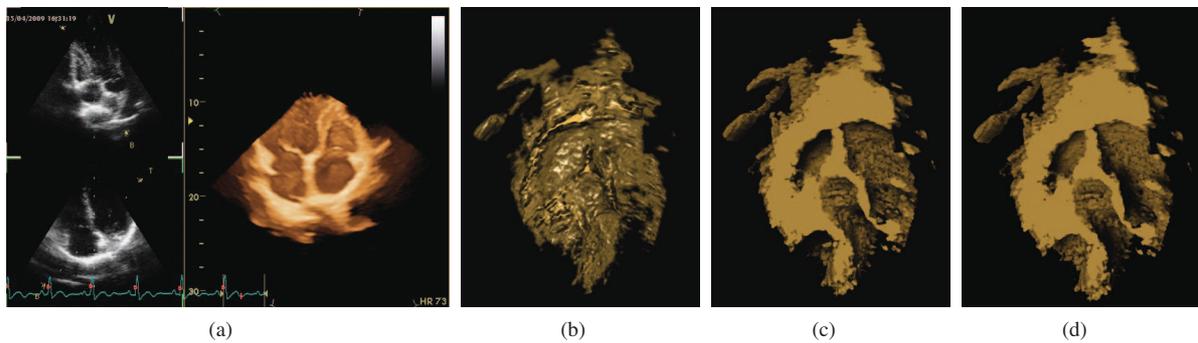


Figure 13: Visualizations of 3D ultrasound of cardiac data: user interface of a 3D cardiac ultrasound workstation (a), a clipped 3D cardiac ultrasound visualization using direct volume rendering and Phong illumination model, rendered with a raycaster (b), clipped 3D cardiac ultrasound visualization using the multidirectional occlusion shading model with light coming from the top left (c) and the bottom left (d).



Figure 14: Visualizations of the bonsai dataset: slice-based volume rendering using a view-aligned slice stack (a) and using a half-angle-aligned slice stack (b).

ics memory, an Intel®Core i7 CPU with 3.07GHz and 12GB of RAM. We measured the performance of our implementation using the gecko dataset of resolution $512 \times 512 \times 88$ voxels, 0.5 voxels sampling distance and viewport-resolution 768×407 pixels. We achieved interactive frame-rates of 19Hz with using the MS-DOS and 18Hz with mul-

tidirectional OS using a 37° angle of aperture while interacting with the viewing parameters. During interaction with the light source, which required update of the kernel, we achieved 14Hz frame-rates. For comparison, using the same framework, a simple slice-based renderer with no shadowing and Phong illumination achieved 25Hz and a high-quality raycaster with Phong illumination and no shadowing achieved 21Hz. We performed the same test with the foot dataset of resolution $256 \times 256 \times 256$ voxels, 0.5 voxels sampling distance and viewport-resolution 531×311 pixels. We achieved 15Hz while using the original MS-DOS approach, 14Hz using our new method, and 12Hz during light source interaction. In this case, a simple slice-based renderer performed at 25Hz and a raycaster at 22Hz. These tests prove that the interactive light source placement is a valuable extension of the original approach traded for a negligible performance penalty.

6. Conclusions

In this paper, we presented a shading model for direct volume rendering, which enables the interactive generation of high-quality soft shadow effects without the need for pre-computation. Our method extends a previous technique to enable interactive placement of the light source. Using elliptic instead of circular footprints, we achieve almost the same performance while greatly improving the flexibility of the method. Additionally, we discussed several applications of such a shading model and consulted a professional illustrator to confirm the importance of freely modifying the light direction.

7. Acknowledgments

This work was carried out within the IllustraSound research project (# 193180), which is funded by the VERDIKT program of the Norwegian Research Council with support of the MedViz network in Bergen, Norway. The authors wish to thank the certified medical illustrator Kari Toverud for consulting, numerous respondents for their feedback and anonymous reviewers for their comments.

References

- [BBC83] BERBAUM K., BEVER T., CHUNG C. S.: Light source position in the perception of object shape. *Perception* 12, 5 (1983), 411–416. 1
- [BG05] BRUCKNER S., GRÖLLER M. E.: VolumeShop: An interactive system for direct volume illustration. In *Proceedings of IEEE Visualization* (2005), pp. 671–678. 5
- [BG07] BRUCKNER S., GRÖLLER M. E.: Enhancing depth-perception with flexible volumetric halos. *IEEE Transactions on Visualization and Computer Graphics* 13, 6 (2007), 1344–1351. 3
- [BLK00] BRAJE W., LEGGE G., KERSTEN D.: Invariant recognition of natural objects in the presence of shadows. *Perception* 29, 4 (2000), 383–398. 1
- [BR98] BEHRENS U., RATERING R.: Adding shadows to a texture-based volume renderer. In *Proceedings of IEEE Symposium on Volume Visualization 1998* (1998), pp. 39–46. 2
- [Bun05] BUNNEL M.: *GPU-Gems*, vol. 2. Addison Wesley, 2005, ch. Dynamic Ambient Occlusion and Indirect Lighting. 2
- [DEP05] DESGRANGES P., ENGEL K., PALADINI G.: Gradient-free shading: A new method for realistic interactive volume rendering. In *Proceedings of Vision, Modeling, and Visualization 2005* (2005), pp. 209–216. 3
- [DSDD07] DACHSBACHER C., STAMMINGER M., DRETTAKIS G., DURAND F.: Implicit visibility and antiradiance for interactive global illumination. *ACM Transactions on Graphics* 26, 3 (2007), 61.1–61.10. 2
- [HKS06] HADWIGER M., KRATZ A., SIGG C., BÜHLER K.: GPU-accelerated deep shadow maps for direct volume rendering. In *Proceedings of SIGGRAPH/EUROGRAPHICS Symposium on Graphics Hardware* (2006), pp. 49–52. 2, 7
- [HLY08] HERNELL F., LJUNG P., YNNERMAN A.: Interactive global light propagation in direct volume rendering using local piecewise integration. In *Proceedings of IEEE/EG International Symposium on Volume and Point-Based Graphics* (2008), pp. 105–112. 3
- [HLY09] HERNELL F., LJUNG P., YNNERMAN A.: Local ambient occlusion in direct volume rendering. *IEEE Transactions on Visualization and Computer Graphics* 99, 2 (2009). 3
- [HYL07] HERNELL F., YNNERMAN A., LJUNG P.: Efficient ambient and emissive tissue illumination using local occlusion in multiresolution volume rendering. In *Proceedings of Volume Graphics 2007* (2007), pp. 1–8. 3
- [KKH02] KNISS J., KINDLMANN G., HANSEN C.: Multidimensional transfer functions for interactive volume rendering. *IEEE Transactions on Visualization and Computer Graphics* 8, 3 (2002), 270–285. 2, 7
- [KMK94] KERSTEN D., MAMASSIAN P., KNILL D.: *Moving cast shadows and the perception of relative depth*. Tech. Rep. 6, Max-Planck-Institut für biologische Kybernetik, Tübingen, Germany, 1994. 1
- [Kne07] KNECHT M.: *State of the art report on ambient occlusion*. Tech. rep., Technische Universität Wien, Vienna, Austria, 2007. 2
- [KPH*03] KNISS J., PREMOZE S., HANSEN C., SHIRLEY P., MCPHERSON A.: A model for volume lighting and modeling. *IEEE Transactions on Visualization and Computer Graphics* 9, 2 (2003), 150–162. 2
- [Lev87] LEVOY M.: Display of surfaces from volume data. *IEEE Computer Graphics and Applications* 8 (1987), 29–37. 2
- [Max95] MAX N.: Optical models for direct volume rendering. *IEEE Transactions on Visualization and Computer Graphics* 1, 2 (1995), 99–108. 2
- [MS09] MÉNDEZ-FELIU À., SBERT M.: From obscurities to ambient occlusion: A survey. *"The Visual Computer"* 25, 2 (2009), 181–196. 2
- [RDR10] ROPINSKI T., DÖRING C., REZK-SALAMA C.: Interactive volumetric lighting simulating scattering and shadowing. In *Proceedings of IEEE Pacific Visualization* (2010), pp. 169–176. 3, 7
- [RMSD*08] ROPINSKI T., MEYER-SPRADOW J., DIEPENBROCK S., MENSMA J., HINRICHS K. H.: Interactive volume rendering with dynamic ambient occlusion and color bleeding. *Computer Graphics Forum* 27, 2 (2008), 567–576. 3
- [RS07] REZK-SALAMA C.: GPU-based Monte-Carlo volume raycasting. In *Proceedings of Pacific Graphics 2007* (2007), pp. 411–414. 2
- [SA07] SHANMUGAM P., ARIKAN O.: Hardware accelerated ambient occlusion techniques on GPUs. In *Proceedings of Symposium on Interactive 3D Graphics and Games* (2007), pp. 73–80. 2
- [SPH*09] SCHOTT M., PEGORARO V., HANSEN C., BOULANGER K., STRATTON J., BOUATOUCH K.: A directional occlusion shading model for interactive direct volume rendering. *Computer Graphics Forum* 28, 3 (June 2009), 855–862. 1, 3
- [Ste03] STEWART A. J.: Vicinity shading for enhanced perception of volumetric data. In *Proceedings of IEEE Visualization 2003* (2003), pp. 355–362. 3
- [YKZ91] YAGEL R., KAUFMAN A., ZHANG Q.: Realistic volume imaging. In *Proceedings of IEEE Visualization 1991* (1991), pp. 226–231. 2
- [ZIK98] ZHUKOV S., IONES A., KRONIN G.: An ambient light illumination model. In *Rendering Techniques* (1998), pp. 45–56. 2