Performance and Quality Analysis of Convolution-Based Volume Illumination

Paolo Angelelli
University of Bergen
Bergen, Norway
Paolo.Angelelli@UiB.no

Stefan Bruckner
University of Bergen
Bergen, Norway
Stefan.Bruckner@UiB.no

ABSTRACT

Convolution-based techniques for volume rendering are among the fastest in the on-the-fly volumetric illumination category. Such methods, however, are still considerably slower than conventional local illumination techniques.
In this paper we describe how to adapt two commonly used strategies for reducing aliasing artifacts, namely pre-integration and supersampling, to such techniques. These strategies can help reduce the sampling rate of the lighting information (thus the number of convolutions), bringing considerable performance benefits. We present a comparative analysis of their effectiveness in offering performance improvements. We also analyze the (negligible) differences they introduce when comparing their output to the reference method.
These strategies can be highly beneficial in setups where direct volume rendering of continuously streaming data is desired and continuous recomputation of full lighting information is too expensive, or where memory constraints make it preferable not to keep additional precomputed volumetric data in memory. In such situations these strategies make single pass, convolution-based volumetric illumination models viable for a broader range of applications, and this paper provides practical guidelines for using and tuning such strategies to specific use cases.

Keywords

Volume Rendering, Global Illumination, Scientific Visualization, Medical Visualization

1 INTRODUCTION

In recent years different medical imaging technologies, such as computed tomography, ultrasonography and microscopy [5], became capable of generating real-time streams of volumetric data at high frame rates. To visualize such data, volume raycasting [2, 10], capable of displaying surfaces from volumetric data without preprocessing, is often used. This happens in particular in situations where inspection of the acquired data is useful already during the acquisition, such as in 4D Echography where volume rendering of real-time data is employed even for guiding interventions. In these cases conventional direct volume rendering techniques that employ local illumination models are generally used, as they are efficient enough to keep up with the incoming data rate when executed on modern GPU hardware, even when not high end. However, just like in polygonal rendering, rendering volume data using an illumination model that approximates global illumination better than simple local shading models is important for numerous reasons, as recent user studies have demonstrated [11, 17]. Researchers have therefore been very active in the last years in proposing efficient and realistic approximations of global illumination, comprehensively covered in a recent survey by Jonsson et al. [7]. Despite the advances in this field, volumetric illumination methods that offer the best performance rely on expensive preprocessing steps to speed up the rendering by reusing precomputed information. Such preprocessing is not applicable in a number of situations, like, for example, when the volume data to be rendered change continuously, but also when memory constraints (e.g., in the case of portable devices or large datasets) make it preferable not to store an additional precomputed illumination volume.

There is, however, a category of techniques that approximate volumetric lighting (single and sometimes multiple scattering) in the same pass used to generate the image, without the need for preprocessing or storing the whole illumination volume. Nonetheless even the fastest methods in this category are on average six to eight times slower [18, 13] than conventional GPU-based direct volume rendering methods using ray-casting and local illumination models such as Phong shading. This performance penalty can be a serious issue where there are constraints on the computational capacity of the system, or when the rendering pipeline includes additional computationally expensive stages such as volume denoising.

In this paper we focus on convolution-based volumetric illumination models [8, 9, 15, 16, 13], a subcategory of single pass volumetric illumination methods built upon slice-based rendering, that operate by iteratively diffusing the lighting information slice after slice using convolutions. Since the geometry setup using ping-pong buffers is a costly operation and, moreover, the convo-

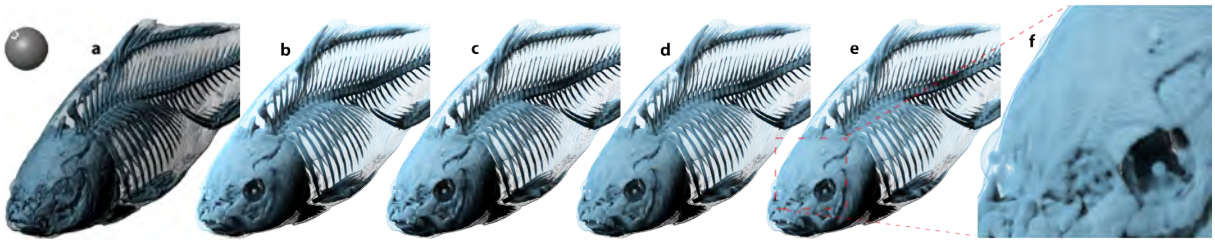

Figure 1: Volume rendering of the carp dataset. **(a)** Raycasting using Phong shading. **(b)** Instant convolution shadows (ICS) with sampling distance of 0.33 voxels (reference). **(c)** ICS with sampling distance of 1 voxel. **(d)** Supersampled convolution shadow (SCS) with a slice distance of 1.3 voxel and 4 tissue subsamples. **(e)** SCS with a slice distance of 1.5 voxels and 5 subsamples (tissue sampling distance of 0.25voxel). **(f)** A closeup highlighting how 1.5 voxels slice distance introduces aliasing artifact despite the dense tissue supersampling. The SCS method, however, allows to increase the inter-slice distance considerably, with an almost linear performance increase. Computation times from left to right: 43ms, 202ms, 88ms, 84ms, 79ms .

lution is performed for every pixel of the view-aligned slices, the sampling distance (and thus the number of slices used for the rendering) and the time necessary for rendering every frame are linearly dependent.

In this paper we analyze the impact of the sampling distance on the performance of this approach in generating aliasing-free images, and incorporate and evaluate the effect of two commonly used strategies to lower this distance: pre-integration [3] and supersampling. The contribution of this paper is therefore twofold. First, we introduce two methods to adapt pre-integration and volume supersampling to convolution-based volumetric illumination techniques, which allow decoupling the sampling rates of the lighting information from the one of the volume. Then we provide a quantitative evaluation of the effects that these strategies have on the performance and practical guidelines for choosing algorithm parameters in order to achieve the best performance without compromising the image quality. We demonstrate that using such strategies can lead to considerable speedups (over 170% in the average case) compared to the standard convolution-based illumination, and, in certain cases, can achieve performance comparable to conventional local illumination methods (see Figure 1 for an example). These performance gains can be instrumental in bringing advanced illumination to volume rendering of streaming data, especially on computationally limited devices, or where the compute unit is used for other computationally expensive steps which are required for the rendering. These strategies can also be beneficial in presence of static data but when, for example, the amount of graphics memory is limited, and precomputing volumetric light information is not preferrable.

2 RELATED WORK

In the area of interactive volume rendering different lighting models to approximate global illumination have been proposed. A thorough overview of such techniques has been provided by Jonsson et al. [7]. In their survey, the authors classify the various techniques in five categories: *local-region-based*, *slice-based*, *light-space-based*, *lattice-based* and *basis-function-based*. Each of these categories describe the underlying paradigm used for calculating volumetric lighting information. The authors also provide a comprehensive analysis of the individual methods, their memory requirements, and their computational costs. The computational costs have been further subdivided into the cost for rendering an image, and the cost for updating the data, the transfer function or the light direction.

For scenarios in which the data is continuously varying we are mostly interested in whether the total time necessary to render the data for the first time exceeds the data rate or not. We therefore adopt a simpler classification here, depending on whether a method requires substantial pre-computation or whether it can produce the final image at interactive frame rates calculating the illumination information on-the-fly. We refer to Jonsson et al. with respect to methods that fit the first of these two classes. In the second class we have splatting-based methods, slice-based methods, and image-plane-sweep-based methods. Splatting was extended to support volumetric lighting by Nulkar and Mueller with the shadow splatting method [12]. This method require an additional pass and the storage of the shadow volume, so it is not an on-the-fly method. However, Zhang and Crawfis [19, 20] later extended the method relaxing these constraints. Still, splatting remains more suitable for sparse or unstructured grids than for dense cartesian grids.

Most of the work in on-the-fly volume illumination can be found in the slice-based category, since synchronization is one of the main issues in calculating the light propagation, and performing slice-based volume rendering implicitly synchronizes the ray front, simplifying the problem. The first method introducing volumetric lighting using this rendering paradigm was *half-angle slicing*, presented by Kniss et al. [8, 9]. The key

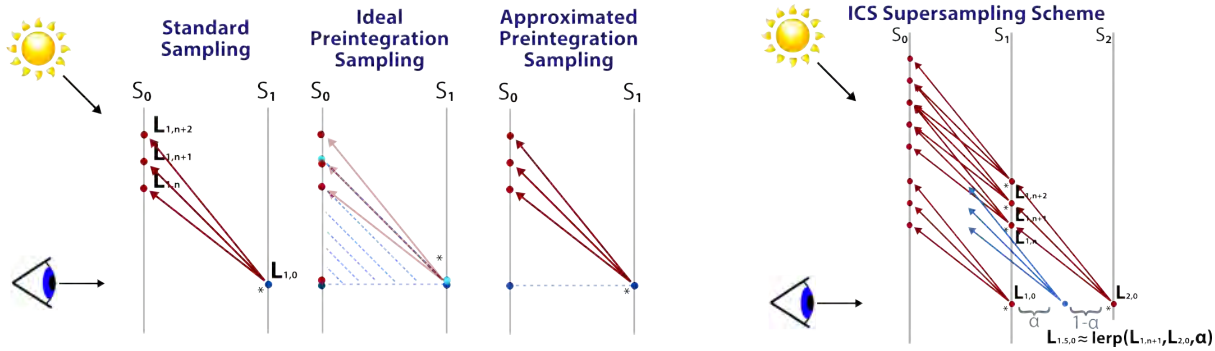

Figure 2: Illustration of our modified schemes for pre-integration (left) and supersampling (right). Correct pre-integration should include in the lookup table also the entry and exit light value. We propose to approximate it performing only tissue pre-integration and sampling the light at S1 position. We also propose to use linearly interpolated light values from the two previous light buffers to calculate the illumination for supersampled tissue samples.

concepts of this method were the implementation of a backward-peaked phase function by iterative convolution and the selection of the slicing direction half-way (hence the name) between viewing and light direction. Schott et al. [15] later presented the *directional occlusion shading* method, constrained to headlight setups to use view-aligned slices, and the same technique to implement the backward-peaked phase function via iterative convolution. However, unlike half-angle slicing, directional occlusion shading does not need two rendering passes per slice. This method was later extended by Šoltészová et al. [16], to allow variable light directions while keeping view-aligned slices. The authors called it *multidirectional occlusion shading*, and also illustrated the advantages of using view-aligned slices in terms of image quality as opposed to half-angle slicing. This method was further improved by Patel et al. [13] with their *instant convolution shadows* method, by using an optimized convolution kernel and allowing the integration of polygonal geometry, making it suitable for volumetric detail mapping to geometrical models.

In the last category, the first and currently only method presented was by Sunden et al. [18], with the *image plane sweep* volume illumination technique. In this method ray-casting is chosen over slice-based rendering, and the rays are not traversed simultaneously, but serialized in a sweep over the image plane. The sweep direction is dependent on the light direction so that the ray direction is orthogonal to the light and subsequent rays can make use of light contributions from previous rays. In their paper the authors show that the performance of their method is similar to half-angle slicing. In this work we focus on slice-based iterative convolution methods.

The last aspect to discuss is how to analyze the results of volume rendering techniques. One of the goals that we have in this work is to improve performance while maintaining the generated images free of aliasing. We identify the optimal parameter setting for the different sampling distances (that is, the most efficient setting that yield aliasing-free images) in a qualitative manner. However, quantitative theoretical models to evaluate the amount of error in volume rendering due to discretization also exist, like the one proposed by Etiene et al. [4], or the method to determine proper sampling frequency of function compositions proposed by Bergner et al. [1]. Performance-wise it has been a common practice to compare different methods on the same viewport size, sampling distance and transfer function, averaging the rendering times over several frames from different viewing direction [14, 18]. In this work we adopt the same strategy. *Timings are averaged over several frames and the viewport size is always fixed to 512x512 pixels*.

3 METHOD

To explain how to adapt supersampling and pre-integration for a convolution-based volumetric illumination model, we can use the Instant Convolution Shadow (ICS) method [13] as the reference model. The basic idea of ICS is that each volume sample on a slice acts as light occluder but also as shadow receiver. This means that every sample which, after classification, maps to a non-fully transparent color, will cast shadows onto the next slice. To compute the amount of light that is transmitted from slice n to a position on slice $n+1$, the incoming light on slice n is first attenuated by the opacity of the samples on slice n, and then this outgoing light is convolved with a kernel $k(x)$. This operation is iterated for every pixel on every slice, and the iterative process propagates the lighting information to the end of the scene.

3.1 Pre-integrated ICS

Pre-integration [3] works by assuming linear variation between two consecutive volume samples. It is then

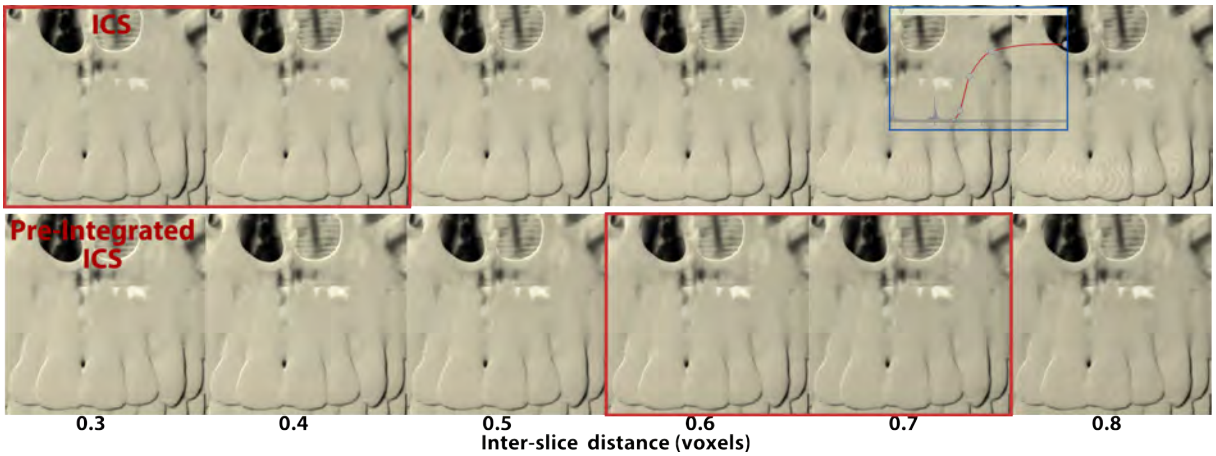

Figure 3: Assessment of the largest sampling distance to produce aliasing-free images for one scene. The transitions where noticeable aliasing appeared are shown in red. Using pre-integration produced identical images and, as expected, allowed to significantly increase the sampling distance while still preventing aliasing.

possible to precompute the volume rendering integral between all possible combination of data values, and store it in a 2D lookup table. During rendering, a simple 2D texture lookup is used. In practice, this approach enables to use of much higher sample distances without noticeable artifacts [3]. However, the basic pre-integration method does not consider illumination, as the resulting increase in dimensionality of the lookup table would make the approach impractical. Previous work [6] showed how to combine local gradient shading with pre-integration by combining two 2D look-up tables. In case of non-local volumetric lighting this is not possible, as the light information depends on the neighborhood of a fragment (see Figure 2).

For this reason we suggest to use standard pre-integration and ignore lighting in the pre-computation. This requires only the conventional 2D lookup table. In this approximation the light propagation proceeds as in the conventional ICS, but the opacity used to attenuate the light comes from the pre-integrated value. We analyze the effect that this approximation has on the image quality, and to what extent it allows us to reduce the inter-slice distance in Section 5.

3.2 Supersampled ICS

The second strategy to increase the distance between slices (and hence, the number of convolutions performed), while still sampling the volumetric function at a sufficiently high rate is to acquire additional volume samples between consecutive slices. The rationale behind this approach is that the color and opacity contributions between consecutive slices are still taken into account, but the illumination propagation is performed at a lower frequency. Such a strategy has pros and cons as compared to pre-integration, where the color is calculated using a finer integration step, but on approximated scalar field values, varying linearly between the front and the back sample. However, these two strategies can also be combined. In order to adapt supersampling to a slice-based renderer with convolution-based lighting, it is necessary to define what light contribution these additional samples collected in between two subsequent slices should receive. The correct solution is illustrated in In Figure 2 on the right (blue convolution). Since this convolution is not possible to calculate due to missing data, we propose an approximation scheme for the light contribution on the additional samples by using their position α in between the slices (see Figure). We then linearly interpolate the light contribution of the current and previous light using this position as the weight.

4 TECHNICAL REALIZATION

Both of these strategies have been shown to be effective in reducing aliasing artifacts, indirectly allowing larger sampling distances. In the specific case of volumetric lighting by convolution shadows, our proposed adaptations blend in the algorithm and are compatible with additional features such as variable light direction, multiple light sources (which can greatly benefit from lower sampling distances), non-white lights or chromatic shadows.

To quantify the benefits that pre-integration and supersampling can provide, we integrated them into a reference implementation of the ICS method. We chose this method because it introduces a number of optimizations over similar methods previously published [16, 15], both from a performance and from an image quality point of view, as discussed in Section 2. The ICS method can be therefore considered one of the most efficient convolution-based volumetric shadows techniques available at the moment.

The necessary adaptations consists of two main ingredients: a loop in the fragment shader to collect the ad-

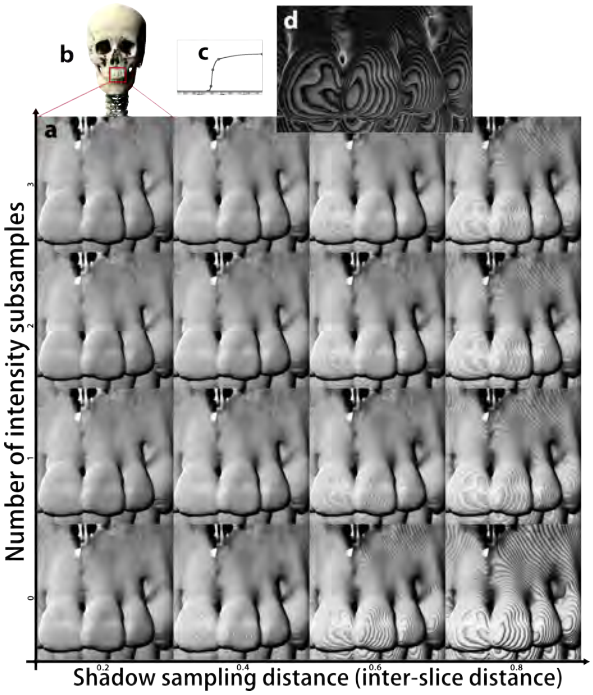

Figure 4: **(a)** Evaluation of the 2D parameter space for supersampled ICS. On the x-axis the inter-slice distance and on the y-axis the number of subsamples are shown. Due to the integral number of possible subsamples, we use x-increments of 0.2 voxels to keep the volume sampling distance identical on the diagonal. Note how increasing the number of substeps does not prevent aliasing anymore after exceeding a certain slice distance. Note that the zoomed views have been desaturated and auto-leveled to enhance the aliasing artifacts, making them easier to see in print. **(b)** Rendering of the whole dataset. **(c)** The transfer function used to geneate these images (same as in Figure 3. **(d)** Absolute differences between the bottom left and the bottom right view (multiplied by a factor of 10 for better visibility). Quantitative measurements are given in Table 1.

ditional samples and an additional color attachment to carry ahead the value of 2 light buffers. However, it should be noted that, if we discard refraction effects that change the color of the light when it propagates in the media, the additional color attachment is not necessary as the light attenuation, even for non-white light, could be approximately described by a single scalar value.

5 RESULTS

5.1 Analysis setup

We carried out a thorough analysis of the different ICS compositing strategies in order to obtain quantitative performance results. To analyze the speedup that these strategies have, we used the average frame rendering time over 100 frames from different view points for different illumination techniques. We compared conventional ICS, ICS with supersampling only for the volume, which from now on will be referred to as Supersampled Convolution Shadows (SCS), pre-integrated ICS and pre-integrated SCS. As a baseline, we also included a conventional volume ray caster with and without local illumination (Phong shading) in the comparison. We conducted our experiments using five different dataset/transfer function combinations. These were a CT dataset of a carp (see Figure 1), a CT dataset of a human head, used with two different transfer functions, one to reveal the skin and one to reveal the skeleton, a CT dataset of a human abdomen revealing the skeleton and the vessels due to contrast agent, and finally a cardiac ultrasound dataset. The dimensions of these volumes are given in Figure 5. The goal of this analysis was to evaluate the performance of each of these techniques in producing artifact-free images. We ran the tests on a workstation equipped with an Intel Core2Quad 2.5GHz CPU, 12GB of RAM and an nVidia Quadro K5000 GPU with 4GB of VRAM. The size of the viewport was fixed to 512x512 pixels.

# Samples / Distance	0	1	2	3
0.2	569ms	572ms	575ms	578ms
0.4	311ms	312ms	314ms	315ms
0.6	225ms	225ms	226ms	227ms
0.8	180ms	181ms	181ms	182ms
0.2	0.0	0.0033	0.0041	0.0045
0.4	0.0060	0.0030	0.0034	0.0035
0.6	0.0118	0.0065	0.0055	0.0054
0.8	0.0187	0.0102	0.0089	0.0083

Table 1: Performance and error analysis for Fig.4. The first table illustrates the necessary time to generate a frame. The second table shows the average pixel difference between the image in the bottom left corner and every other. Pixels have normalized values in the [0,1] interval.

5.2 Parameter Space

We designed the analysis as a two-stage process. In the first analysis stage we estabilished the largest sampling distance for the intensity volume that would still produce aliasing-free pictures using the raycaster, the ICS renderer and the pre-integrated ICS renderer, and used this parameter later on as reference in the performance measurements. This distance was not always the same for the raycasting technique and the ICS technique (slice-based), as these two methods exhibit different aliasing patterns. In particular, and as expected, pre-integrated ICS could consistently tolerate a larger inter-slice distance, which provide an advantage over standard ICS in terms of performance (see Figure 5). This distance was also dependent on the dataset and the transfer function used, so we defined it separately

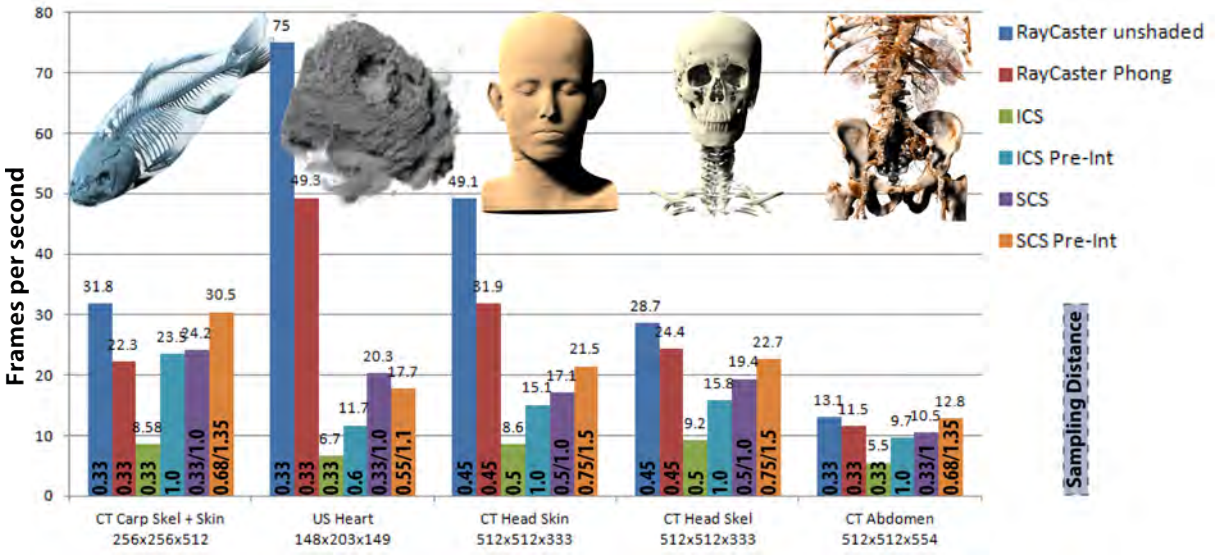

Figure 5: Performance comparison between different rendering methods for five different scenes, depicted on top of each group. On the bottom the size of the volumes in voxels.

for each scene. Figure 3 exemplifies this step for one of the five analyzed scene, in which we qualitatively assessed the larger inter-slice distance that would provide aliasing-free results (for methods to quantitatively assess the amount of aliasing in a rendered image see Section 2).

After the baseline inter-slice distance was identified, we generated the reference images for each of the scenes. In the second step of the analysis we explored the 2D parameter space for the SCS method, in which one dimension is the the inter-slice distance (or the volumetric illumination sampling distance), and the other is the volume sampling distance. However, since our method for integrating supersampling into convolution-based techniques is not able to freely decouple these two parameters (we can only use an integer number of equidistant subsamples between two consecutive sampling slices), we decided to use the number of subsamples as the second parameter in this space. The volume sampling distance can be determined using the formula $SampleDistance = \frac{SliceDistance}{n.o.Subsamples+1}$. Figure 4 shows the result of this exploration for one particular scene using non-preintegrated SCS. This stage was meant to identify the setting of these two parameters that would enable the generation of images identical to the reference most efficiently. After this second stage, optimal parameters for the raycaster, ICS, SCS, pre-integrated ICS and pre-integrated SCS were available, and the performance measurement described in Section 5.1 were conducted using the determined values.

5.3 Analysis results

Figure 5 illustrates the performance that each technique is able to achieve in producing aliasing-free images. When comparing to standard ICS, these results show

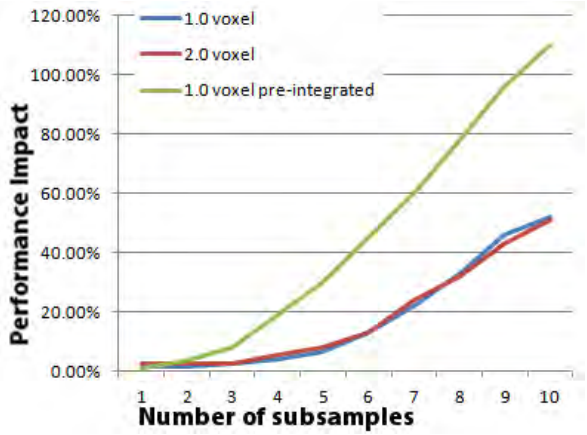

Figure 6: Chart of the performance impact with increasing number of subsamples. In our experiments the slice distance did not play a role, but using pre-integration caused the performance to drop much faster, while regular supersampling comes almost for free for up to 3-4 subsamples.

an average performance increase of 137% for SCS. The worst case scenario for the SCS method has been the CT abdominal scene, where it could offer only a 90% speed increase. In other scenes, in particular in presence of sharper transfer functions such as with the carp dataset or the cardiac ultrasound dataset, the performance increase exceeded 200%.

When using pre-integration, the performance increase over standard ICS is slightly lower despite the usage of same inter-slice distance as SCS in most cases, and even the gathering of only one additional sample as compared to standard ICS (versus the two or three of the SCS method). This behavior can be explained by the fact that sampling a 2D pre-integration table is

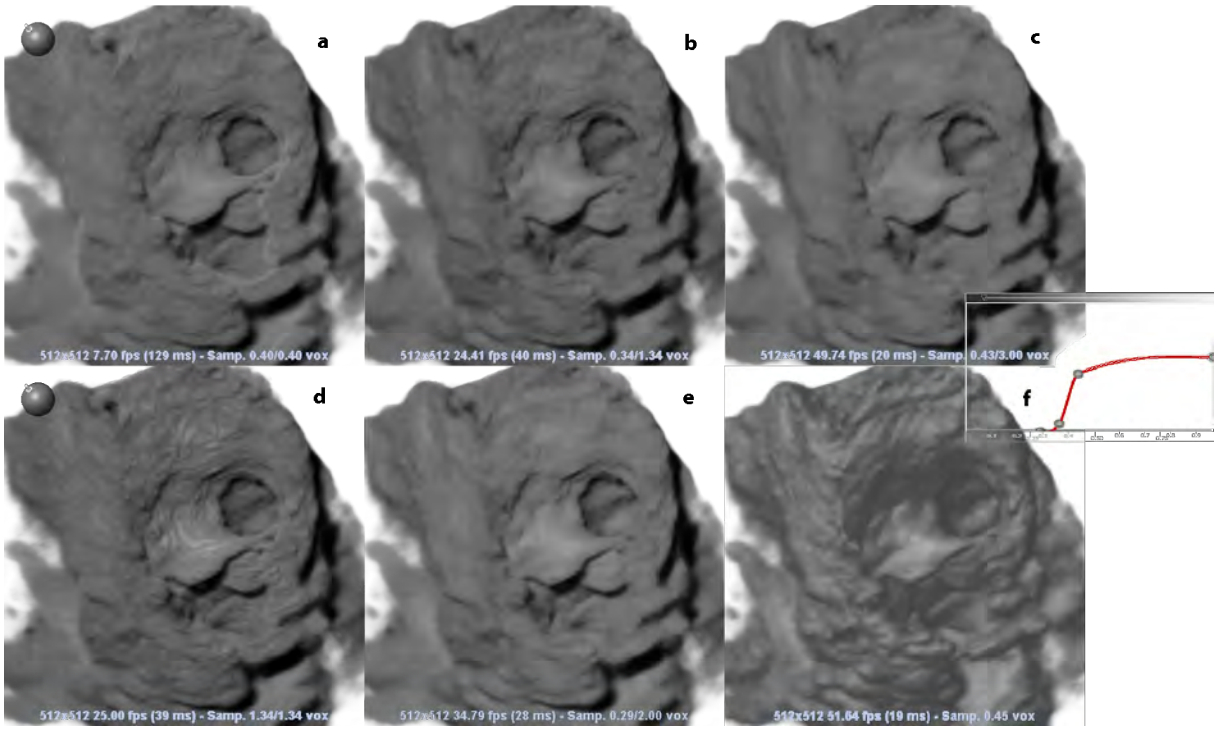

Figure 7: Effect of supersampling a cardiac ultrasound dataset. **(a,d)** ICS with different slice distances. **(b,e,c)** SCS with 1.34 , 2 and 3 voxel slice distances. **(f)** Phong shading for comparison. Note how shadow details on the surfaces progressively disappear with increased sampling distances while shadows casted far away remain the same.

more costly, as the plot in Figure 6, which graphs the penalty for each additional sample for both SCS and pre-integrated SCS, also shows.

Finally, when using both pre-integration and supersampling we could increase the inter-slice distance further without causing aliasing or getting noticeable artifacts in the shading. This combination almost always provided the best performance, except for the cardiac ultrasound dataset, where the inter-slice distance for pre-integration could not be increased as much as in the other scenes. From this analysis we could conclude that, in the average case, the volumetric lighting sampling frequency can be at least halved, when compared to tissue sampling frequency. This possibly due to the lower frequency of the illumination function compared to the post-classified volumetric data. Furthermore we also noticed that the ratio of shadow sampling distance / tissue sampling distance can be further increased in presence of sharper transfer functions.

6 DISCUSSION AND CONCLUSION

Convolution shadow methods and other single pass volumetric illumination techniques can be the only viable option to enable volumetric illumination in a number of application scenarios like real-time 4D echography. Such methods are however constrained on the volume sampling rate by the distance between consecutive slices, requiring a high number of slices for transfer functions containing high frequencies, which consumes a large amount of off-chip GPU memory bandwidth, impacting negatively on the performance. In this work we showed that, by decoupling the sampling rate of the volume from the one of the illumination, we can exploit the fact that illumination is typically less sensitive to lower sampling rates.

We adapted and analyzed two techniques, pre-integration and supersampling, to lower the inter-slice distance and, with some constraints, decouple the two sampling rates. We showed how decoupling these two sampling rates allows less frequent costly convolution operations, bringing a substantial performance increase.

We also discovered that the performance increase using this strategy grows with steeper transfer functions. Both of the strategies analyzed in this paper proved effective, and the most interesting aspect is that, except for one case, they work better when combined. We also experienced that, in certain situations (see Figure 7 for an example), lowering the inter-slice distace beyond what produces images identical to the reference does not immediately introduce aliasing, but the quality of the shading decreases and differences become noticeable. This could however be an acceptable compromise in some situations, in exchange of an additional performance gain.

7 REFERENCES

[1] Steven Bergner, Torsten Moller, Daniel Weiskopf, and David J Muraki. A spectral analysis of function composition and its implications for sampling in direct volume visualization. *Visualization and Computer Graphics, IEEE Transactions on*, 12(5):1353–1360, 2006.

[2] Robert A Drebin, Loren Carpenter, and Pat Hanrahan. Volume rendering. In *ACM Siggraph Computer Graphics*, volume 22, pages 65–74. ACM, 1988.

[3] Klaus Engel, Martin Kraus, and Thomas Ertl. High-quality pre-integrated volume rendering using hardware-accelerated pixel shading. In *Proceedings of the ACM SIGGRAPH/EUROGRAPHICS workshop on Graphics hardware*, pages 9–16. ACM, 2001.

[4] Tiago Etiene, Daniel Jonsson, Timo Ropinski, Carlos Scheidegger, Joao Comba, L Nonato, R Kirby, Anders Ynnerman, and C Silva. Verifying volume rendering using discretization error analysis. *Visualization and Computer Graphics, IEEE Transactions on*, 20(1):140–154, Jan 2014.

[5] Anthony W. P. Fitzpatrick, Sang Tae Park, and Ahmed H. Zewail. Exceptional rigidity and biomechanics of amyloid revealed by 4d electron microscopy. *Proceedings of the National Academy of Sciences*, 110(27):10976–10981, 2013.

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

[7] Daniel Jönsson, Erik Sundén, Anders Ynnerman, and Timo Ropinski. A survey of volumetric illumination techniques for interactive volume rendering. In *Computer Graphics Forum*. Wiley Online Library, 2013.

[8] Joe Kniss, Simon Premoze, Charles Hansen, and David Ebert. Interactive translucent volume rendering and procedural modeling. In *IEEE Visualization 2002*, pages 109–116. IEEE, 2002.

[9] Joe Kniss, Simon Premoze, Charles Hansen, Peter Shirley, and Allen McPherson. A model for volume lighting and modeling. *Visualization and Computer Graphics, IEEE Transactions on*, 9(2):150–162, 2003.

[10] Marc Levoy. Display of surfaces from volume data. *Computer Graphics and Applications, IEEE*, 8(3):29–37, 1988.

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

[12] Manjushree Nulkar and Klaus Mueller. Splatting with shadows. In *Proceedings of the 2001 Eurographics conference on Volume Graphics*, pages 35–50. Eurographics Association, 2001.

[13] Daniel Patel, Veronika Šoltészová, Jan Martin Nordbotten, and Stefan Bruckner. Instant convolution shadows for volumetric detail mapping. *ACM Transactions on Graphics*, 32(5):154:1–154:18, 2013.

[14] Timo Ropinski, C Doring, and Christof Rezk-Salama. Interactive volumetric lighting simulating scattering and shadowing. In *Pacific Visualization Symposium (PacificVis), 2010 IEEE*, pages 169–176. IEEE, 2010.

[15] Mathias Schott, Vincent Pegoraro, Charles Hansen, Kévin Boulanger, and Kadi Bouatouch. A directional occlusion shading model for interactive direct volume rendering. In *Computer Graphics Forum*, volume 28, pages 855–862, 2009.

[16] Veronika Šoltészová, Daniel Patel, Stefan Bruckner, and Ivan Viola. A multidirectional occlusion shading model for direct volume rendering. *Computer Graphics Forum*, 29(3):883–891, 2010.

[17] Veronika Šoltészová, Daniel Patel, and Ivan Viola. Chromatic shadows for improved perception. In *Proceedings of the ACM SIGGRAPH/Eurographics Symposium on Non-Photorealistic Animation and Rendering*, pages 105–116. ACM, 2011.

[18] Erik Sundén, Anders Ynnerman, and Timo Ropinski. Image plane sweep volume illumination. *Visualization and Computer Graphics, IEEE Transactions on*, 17(12):2125–2134, 2011.

[19] Caixia Zhang and Roger Crawfis. Volumetric shadows using splatting. In *Visualization, 2002. VIS 2002. IEEE*, pages 85–92. IEEE, 2002.

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