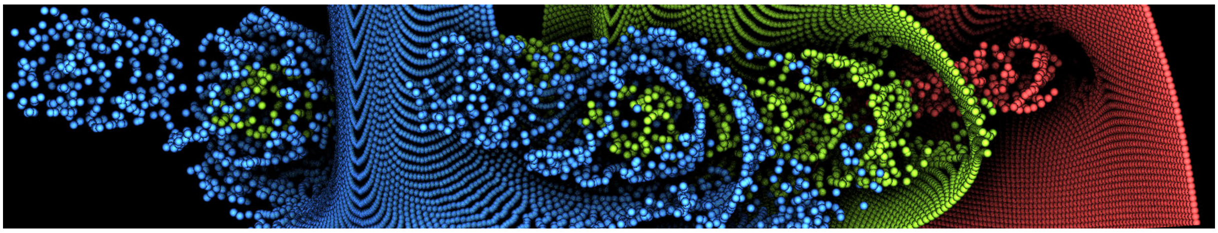
Eurographics Conference on Visualization (EuroVis) (2013)
M. Hlawitschka and T. Weinkauf (Editors)

Short Papers

PointAO — Improved Ambient Occlusion for Point-based Visualization

Sebastian Eichelbaum[1] and Gerik Scheuermann [1] and Mario Hlawitschka [2]

[1]Universität Leipzig, Bild- und Signalverarbeitung, Leipzig, Germany
[2]Universität Leipzig, Wissenschaftliche Visualisierung, Leipzig, Germany



Abstract

The visualization of large amounts of particles, glyphs, and other point-based data plays an important role in many fields of science, among them flow mechanics, molecular dynamics, and medical imaging. The proper perception of spatial structures and spatial relations in the data is crucial to the understanding. To accommodate this aspect, we utilize and improve an existing ambient occlusion approach, originally tailored towards line rendering and extend the approach to be applicable to point-based visualizations.

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

In many fields of science, the visualization of large amounts of particles, glyphs and point-based data plays an important role. Typically, these points and particles have a direct physical meaning, and the proper perception of their global spatial relations and local structure are crucial to understanding the data and the underlying models. It is known that global lighting effects are very important for determining an object's position and spatial relations [Wan92, WFG92, Ram88, LB00]. Hence, global illumination became popular in recent years, especially in molecular data visualization. The goal is to improve the spatial perception and recognition of features in the data. Unfortunately, these methods often suffer several limitations like immense precalculation costs, reduced accuracy due to needed simplifications, or limitations towards the simultaneous shading of local *and* global detail.

With this paper, we improve the LineAO method by Eichelbaum et al. [EHS13], which is tailored towards the specific problems in line rendering and its global illumination and shading. Our contribution is an enhanced screen-space based ambient occlusion approach for point data, which overcomes the before-mentioned problems and provides a greatly *improved structural and spatial perception* with *simultaneous depiction of local and global structures* in *real-time*. In the remainder of this paper, we refer to all the different kinds of three-dimensional point data simply as *point data*, since our approach generally works on a large variety of dense point data sets. Our focus lies on enhanced perception of dense, three-dimensional point clouds. Therefore, our method has no advantages for two-dimensional data.

2. Related Work

Particles-based representations of homogeneous and inhomogeneous media are the basis of various simulation meth-

© The Eurographics Association 2013.

DOI: 10.2312/PE.EuroVisShort.EuroVisShort2013.013-017

delivered by EUROGRAPHICS DIGITAL LIBRARY www.eg.org diglib.eg.org

ods. Instead of representing a continuum, the particles describing the media can be visualized directly. For small particles or particles with only an imaginary extend, glyphs are usually used. Basic representations are spheres, but more-complex objects can describe more properties of the data they represent. With the vanishing memory restrictions and data throughput, unstructured point data becomes more and more popular and the possibility for direct visualization of point data often leads to a good first impression of it.

In computer graphics and visualization, there is a wide variety of techniques for providing depth cues and improved global shading. A very general approach is to mask the depth buffer for depth darkening and silhouette shading [LCD06]. Other approaches utilize illustration techniques to improve spatial perception or object layering [EBRI09, vdZLBI11]. In recent years, global illumination, especially ambient occlusion [ZIK98], became more prominent in visualization and particularly in direct volume rendering [RDRS10, LR11]. The basic idea of ambient occlusion (AO) is to use the amount of ambient light *not* reaching a certain point on a surface as a shading factor for this point. For a detailed overview on the theory behind AO, please refer to [ZIK98]. The crucial part is to calculate the amount of light reaching each point on each surface when taking all other surfaces and objects into account. To tackle this problem, there are two major types of ambient occlusion approaches: object-space and screen-space (SSAO). Object-space approaches usually simplify the scene and estimate the amount of occlusion in this simplified version. A major disadvantage of these approaches is that they need pre-computation and, thus, have severe limitations for large or dynamic scenes. Tarini et al. [TCM06], e.g., pre-computes the occlusion factors for each vertex in the scene, making it very expensive for large or dynamic datasets. Grottel et al. [GKSE12] compute the ambient occlusion factor in a volume of reduced resolution.

Screen-space approaches overcome this issue by achieving the AO effect phenomenologically, which means that they approximate the shadowing by sampling a pixel's surrounding in screen-space. These approaches have the advantage of being fast but, depending on the method, tend to be less accurate when compared to object space approaches. In terms of clearness of this paper, we refer the reader to [GKSE12] and [EHS13] for an in-depth overview on available approaches and their limitations.

3. Background

In the following, we provide background information on the approach of Eichelbaum et al. [EHS13], upon which this paper builds, and keep the original notation for easier comparison.

The very foundation of each ambient occlusion technique is the discrete description of ambient light that does *not* reach the point P on a surface with normal n due to occlusion. When all objects are opaque, we only need to know the amount of light being occluded on a hemisphere around P in direction of n. This yields the standard, discrete AO equation (cf. previous work)

$$AO_s(P,n) = \frac{1}{s}\sum_{i=1}^{s}(1 - V(\omega_i, P))\langle\omega_i, n\rangle, \qquad (1)$$

which approximates the ambient occlusion factor at a point of a surface by sampling the surroundings of P hemispherically. ω_i is one of the well-chosen s hemispherical direction samples taken into account. The computationally most expensive part is the function $V(\omega_i, P)$, which describes whether the ambient light is reaching P from direction ω_i. Most SSAO approaches, including LineAO, render the scene including the normal information and depth information for each pixel to a texture. Evaluation of V is then done by sampling around a pixel P and checking whether the sampled pixel is to the front or to the back of P. If it is in front, it occludes light. Depending on the number of samples s and the distribution of the sampling directions ω_i, this can lead to severe artifacts, missed occluders, and makes the result dependent on the scaling of the scene and the sampling distance.

The key element of LineAO is that it computes an average of the AO factors for multiple hemisphere radii. It increases the radius in each distance-level j, while reducing the number of samples on outer shells. Additionally, it modifies the sampling radius depending on the current zoom of the scene to stay consistent and weights the visibility function V by a factor g, which depends on the depth-distance, distance-level, and light properties of the occluder.

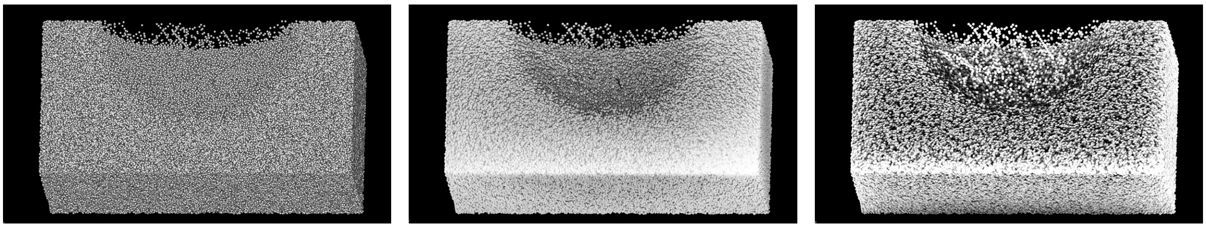
4. Method

To achieve the effects of LineAO for glyph rendering, we need to change three parts of the original algorithm. The following text thereby relates very strongly to Equations 6 and 7 of the original LineAO paper.

Sampling Scheme. LineAO reduces the amount of samples taken into account for each, increasing hemisphere. This can be done since the algorithm uses a Gaussian pyramid for depth- and normal-maps. Thick, more distant bundles merge to single objects in higher levels of the pyramid, thus, reducing the probability to miss them during sampling. This is not the case for glyphs and points. Even dense areas might contain a lot of holes, which do not quickly vanish in higher Gauss-pyramid levels. Hence, there is a need for more samples, even for distant occluders.

Radius Scaling. The original algorithm increased the radius linearly using $r_0 + jz(P)$, where r_0 is a pre-defined minimal radius, $j \geq 0$ the current hemisphere, and $z(P)$ a function denoting the zoom of the scene. In PointAO, we use $\frac{1}{1-d_j(P)} * z(P) * (j^2 + j * r_0)$ as the radius parameter for each hemisphere, where $d_j(P)$ is the depth of the pixel P with d being 1 at the far clipping plane and 0 at the front. The term

© The Eurographics Association 2013.



(a) Phong Shading – 30 FPS (b) SSAO – 19 FPS (c) PointAO – 12 FPS

Figure 1: *Cut through an Argon fluid with enclosed Argon gas bubble with 3 529 344 particles.*

$\frac{1}{1-d_j(P)}$ causes pixels to the front to use a smaller radius, thus, containing more local detail, whereas pixels further back are influenced more by distant occluders. This creates a crisp shading at prominent glyphs in front and a smooth shadow on glyphs in the back.

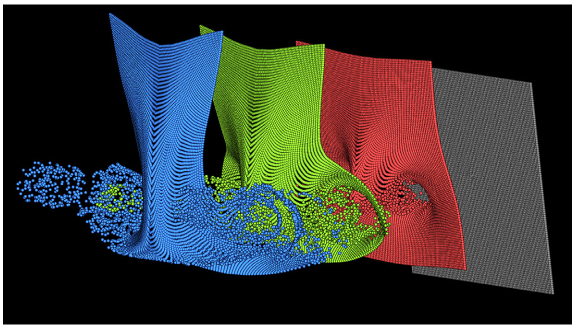
Weighting function. Finally, we modified the weighting function $g_l(\omega, P)$, which weights the occluder influence on P for a given sampling direction ω at a certain distance-level l. It was a combination of depth-based weighting and light-based weighting. The light-based weighting benefits from the fact that bundles of lines merge to surface-like objects with useful normals on them for increasing distance-level. Due to their shape, spherical glyphs scatter the light too much, causing it to be over-estimated in LineAO. Instead, we combine the depth-based weight with $\langle\omega, n_l(P)\rangle$ as in Equation 1 to better retain the glyphs shape in the shading.

Implementation. We have implemented our approach using OpenGL and the render-to-texture functionality, which is the standard way to implement screen-space methods. The concepts are equal to the implementation provided in the original paper. Our implementation is freely available at `http://www.openwalnut.org`.

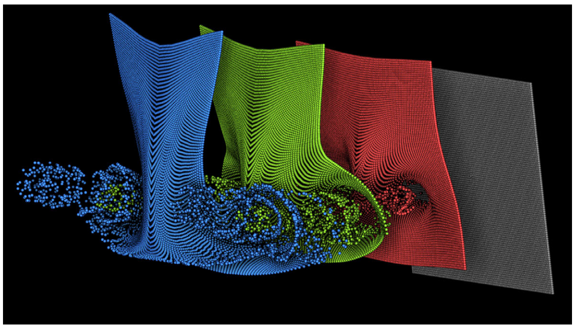
5. Results and Discussion

To demonstrate our technique, we apply the rendering to three types of data. We compare our method to the well-known and widely-used Crytek SSAO [Mit07] approach and Blinn-Phong shaded [Bli77] spherical glyphs. The performance measures where taken on a GeForce GTX 480 mid-range consumer graphics card using FullHD resolution and naive rendering, without any optimization towards occlusion-culling or similar geometry reduction techniques.

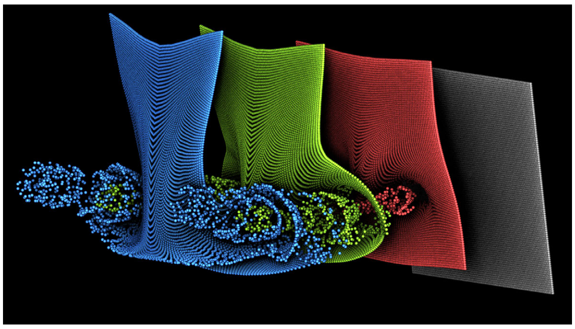
Argon Bubble. Figure 1 shows a cut through a cube of simulated argon fluid, which contains an argon gas bubble. The simulation uses a truncated Lennard-Jones pair [Jon24] potential for the intermolecular repulsion and short-range dispersive attraction. The standard Phong-shaded spherical glyphs allow only to guess that there is a cavity inside the block. Although SSAO clearly shows the cavity, it provides no further local detail due to the very large radius needed.



(a) Phong Shading – 73 FPS



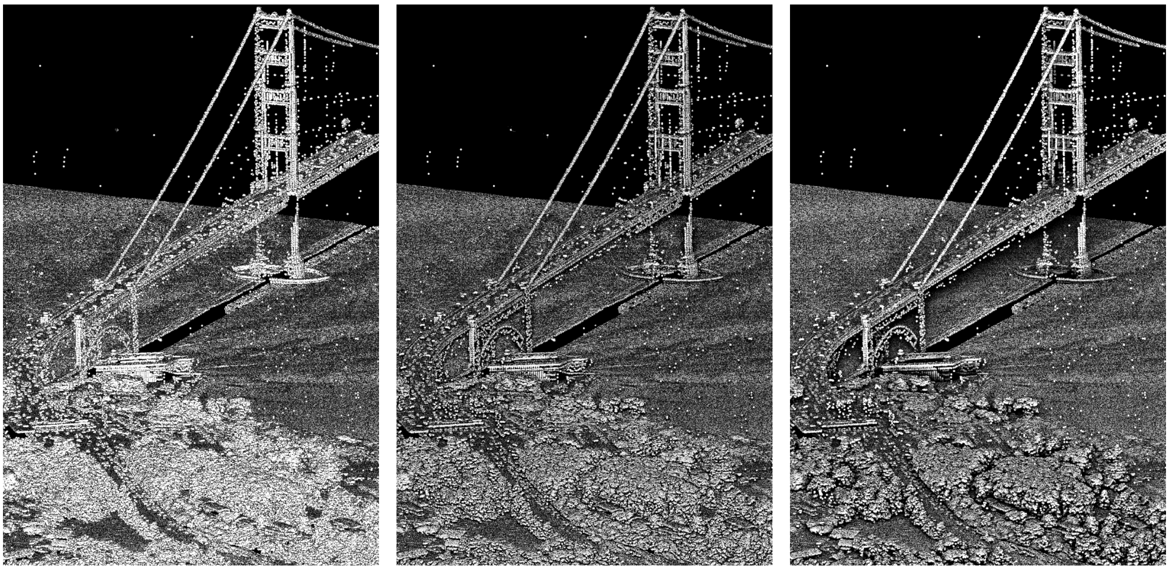
(b) SSAO – 29 FPS



(c) PointAO – 20 FPS

Figure 2: *81 554 Particles flowing into the leading edge vortices of an inclined delta wing.*

© The Eurographics Association 2013.



(a) Phong Shading – 9 FPS (b) SSAO – 7 FPS (c) PointAO – 5 FPS

Figure 3: *Light detection and ranging (LIDaR) scan of the Golden Gate Bridge area with 14 614 901 sample points.*

Our PointAO approach keeps the global spatial information and shows the cavity but also unveils the flowing argon gas atoms, which are not clearly visible in the other approaches.

Flow Particle Data. Figure 2 shows traced particles in the leading edge vortices of an inclined delta wing at four different time-steps. Tracers are seeded on an initial plane (gray) and shown after different time intervals. The simple glyph rendering does not provide any cue on the spatial structure of the particles inside the vortex, nor does it provide any cue for estimating depth distance between the different particle planes and the vortices. The difference between the SSAO and PointAO renderings are rather subtle at a first glance. The SSAO rendering provides a structural cue inside the vortices and a subtle global shading between the planes. This is due to the very solid-geometry alike structure, the well-chosen SSAO radius, and an overemphasized AO factor. Generally, SSAO is much more dependent on well-chosen algorithm parameters, which vary from dataset to dataset. Choosing a smaller radius for SSAO would emphasize the particles inside the vortex, whereas a larger radius would emphasize the spatial relation between the planes and the vortices as a whole. With PointAO, we are able to emphasize local and global structure in equal measure.

LIDaR Scan. Figure 3 shows light detection and ranging (LIDaR) data of the Golden Gate Bride area. It is an imaging technique that acquires millions of points on visible surfaces leading to a point-based reconstruction of objects scanned. Instead of preprocessing the data to obtain surface meshes [ABCO*01] or rendering the data after assigning surface normals [KBK08], we directly display the point data. Especially in the chosen data set, surface-based techniques fail to provide sufficient representations of the bridge or the trees in front of the bridge. Due to the sharp edges of the glyphs, the simple glyph rendering already shows the basic structure of the Golden Gate bride area. The SSAO rendering adds additional, very local shading details due to the chosen, small sampling radius. This can be seen especially in the trees area. Our PointAO approach provides these local shading details as well, but adds more global shadows to the scene, e.g., the shadows between the groups of trees and below the bridge, which are not directly visible in the SSAO image.

6. Conclusion and Future Work

We have presented an improved version of the LineAO algorithm, which is optimized towards real-time rendering of point-based data. It is able to handle arbitrary types of glyphs and does not need any pre-computation, thus, making it ideal for interactive exploration and filtering of data. It allows simultaneous depiction of local *and* global spatial relations and structure in the data, while being consistent under modification and interaction.

7. Acknowledgements

We want to thank Martin Horsch (TU Kaiserslautern) and Markus Rütten (DLR Göttingen) for providing the datasets.

References

[ABCO*01] Alexa M., Behr J., Cohen-Or D., Fleishman S., Levin D., Silva C. T.: Point set surfaces. In *Proceedings of the conference on Visualization '01* (2001), VIS '01, pp. 21–28. doi:10.1109/VISUAL.2001.964489. 4

[Bli77] Blinn J. F.: Models of light reflection for computer synthesized pictures. In *SIGGRAPH '77: Proceedings of the 4th annual conference on Computer graphics and interactive techniques* (1977), pp. 192–198. doi:10.1145/563858.563893. 3

[EBRI09] Everts M. H., Bekker H., Roerdink J. B., Isenberg T.: Depth-dependent halos: Illustrative rendering of dense line data. *IEEE Transactions on Visualization and Computer Graphics 15*, 06 (2009), 1299–1306. doi:10.1109/TVCG.2009.138. 2

[EHS13] Eichelbaum S., Hlawitschka M., Scheuermann G.: LineAO — improved three-dimensional line rendering. *IEEE Transactions on Visualization and Computer Graphics 19*, 3 (2013), 433–445. doi:10.1109/TVCG.2012.142. 1, 2

[GKSE12] Grottel S., Krone M., Scharnowski K., Ertl T.: Object-space ambient occlusion for molecular dynamics. In *Proceedings of IEEE Pacific Visualization Symposium 2012* (2012), pp. 209–216. doi:10.1109/PacificVis.2012.6183593. 2

[Jon24] Jones J. E.: On the determination of molecular fields. ii. from the equation of state of a gas. *Proceedings of the Royal Society of London. Series A 106*, 738 (1924), 463–477. doi:10.1098/rspa.1924.0082. 3

[KBK08] Kreylos O., Bawden G., Kellogg L.: Immersive visualization and analysis of lidar data. In *Advances in Visual Computing*, Bebis G., Boyle R., Parvin B., Koracin D., Remagnino P., Porikli F., Peters J., Klosowski J., Arns L., Chun Y., Rhyne T.-M., Monroe L., (Eds.), vol. 5358 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, 2008, pp. 846–855. doi:10.1007/978-3-540-89639-5_81. 4

[LB00] Langer M., Bülthoff H.: Depth discrimination from shading under diffuse lighting. *Perception 29* (2000), 649–660. doi:10.1068/p3060. 1

[LCD06] Luft T., Colditz C., Deussen O.: Image enhancement by unsharp masking the depth buffer. *ACM Transactions on Graphics 25*, 3 (2006), 1206–1213. doi:10.1145/1179352.1142016. 2

[LR11] Lindemann F., Ropinski T.: About the influence of illumination models on image comprehension in direct volume rendering. *IEEE Transactions on Visualization and Computer Graphics 17* (2011), 1922–1931. doi:10.1109/TVCG.2011.161. 2

[Mit07] Mittring M.: Finding next gen: Cryengine 2. In *ACM SIGGRAPH 2007 courses* (2007), SIGGRAPH '07, pp. 97–121. doi:10.1145/1281500.1281671. 3

[Ram88] Ramachandran V. S.: Perception of shape from shading. *Nature 331* (1988), 163–166. doi:10.1038/331163a0. 1

[RDRS10] Ropinski T., Döring C., Rezk Salama C.: Interactive volumetric lighting simulating scattering and shadowing. In *IEEE Pacific Visualization* (2010), pp. 169–176. doi:10.1109/TVCG.2011.161. 2

[TCM06] Tarini M., Cignoni P., Montani C.: Ambient occlusion and edge cueing for enhancing real time molecular visualization. *IEEE Transactions on Visualization and Computer Graphics 12* (2006), 1237–1244. doi:10.1109/TVCG.2006.115. 2

[vdZLBI11] van der Zwan M., Lueks W., Bekker H., Isenberg T.: Illustrative molecular visualization with continuous abstraction. *Computer Graphics Forum 30*, 3 (2011), 683–690. doi:10.1111/j.1467-8659.2011.01917.x. 2

[Wan92] Wanger L.: The effect of shadow quality on the perception of spatial relationships in computer generated imagery. In *Proceedings of the 1992 symposium on Interactive 3D graphics* (1992), I3D '92, pp. 39–42. doi:10.1145/147156.147161. 1

[WFG92] Wanger L. C., Ferwerda J. A., Greenberg D. P.: Perceiving spatial relationships in computer-generated images. *IEEE Computer Graphics and Applications 12* (1992), 44–51, 54–58. doi:10.1109/38.135913. 1

[ZIK98] Zhukov S., Inoes A., Kronin G.: An Ambient Light Illumination Model. In *Rendering Techniques '98* (1998), Drettakis G., Max N., (Eds.), Eurographics, pp. 45–56. doi:10.1007/978-3-7091-6453-2_5. 2

© The Eurographics Association 2013.