Supplementary Document on Moment-Based Order-Independent Transparency

CEDRICK MÜNSTERMANN, STEFAN KRUMPEN, and REINHARD KLEIN, University of Bonn, Germany
CHRISTOPH PETERS, Karlsruhe Institute of Technology, Germany and University of Bonn, Germany

ACM Reference Format:
Cedrick Münstermann, Stefan Krumpen, Reinhard Klein, and Christoph Peters. 2018. Supplementary Document on Moment-Based Order-Independent Transparency. *Proc. ACM Comput. Graph. Interact. Tech.* 1, 1, Article 7 (May 2018), 4 pages. https://doi.org/10.1145/3203206

If all choices of moments and quantization schemes are taken into account, the main document discusses a total of twelve different moment-based reconstructions for the transmittance. In this supplementary document, we provide a concise overview of the optimal strategies for biasing and the quantization transforms. Besides we provide results for variants of techniques, which have not been discussed in the paper. HLSL code for all of our techniques and a video are available separately.

1 OPTIMAL BIASING

When working with moments, the introduction of rounding errors is inevitable. If a vector of moments $b \in \mathbb{C}^{m+1}$ is corrupted by noise, it may end up being invalid, i.e. there may be no depth distribution Z such that $b = \mathcal{E}_Z(\boldsymbol{b})$. In the algorithms, this manifests as matrices with negative eigenvalues such that the Cholesky decomposition fails. The result are elongated regions with faulty pixels.

The proposed solution in moment shadow mapping [Peters and Klein 2015] is to pull the vector back into the admissible domain through linear interpolation towards a fixed vector of biasing moments $b^\star \in \mathbb{R}^{m+1}$. The biased and normalized vector that is used as input to the reconstruction is

$$(1 - \alpha) \cdot \frac{b}{b_0} + \alpha \cdot b^\star.$$

For trigonometric moments, the natural choice of $b^\star$ is to set $b_0^\star = 1$ and all other entries to zero [Peters and Klein 2015]. This is the only choice that is invariant under periodic shifts of the depth domain. For power moments, a sophisticated optimization procedure has been proposed to find the vector of biasing moments that is most effective in the worst case [Peters et al. 2017]. The result depends on the exact nature of the rounding errors, which in turn depends on the quantization scheme.

The strength of the moment bias $0 < \alpha \ll 1$ is best found empirically by increasing it until the results are free of artifacts. It depends upon the amount of rounding error that is introduced. In other applications of moment-based reconstructions, this depends only on the chosen quantization scheme and not on the scene. As discussed in the paper, this is not true with the additive rendering employed here because rounding errors may accumulate over many passes. Greater overdraw implies greater rounding errors. In practice, this is no problem with single precision floats but does require some extra work when storing moments in 16 bits per pixel.

Table 1 provides the optimal vectors of biasing moments for all techniques as described above. The given values for the moment bias α are values that work well at little overdraw. When using 16 bits per moment, they may need to be increased for scenes with large overdraw.

2 QUANTIZATION AT 16 BITS PER MOMENT

To minimize the introduction of rounding errors, it is advisable to apply an affine transform $\Theta_m^\star$ to the normalized vector of moments before storing it in 16 bits per moment. This transform is chosen to maximize $|\det \Theta_m^\star|$ without violating the valid range of output values $[0, 1]$ [Peters and Klein 2015]. For efficiency reasons, it is additionally chosen to transform odd and even moments separately [Peters et al. 2017].

Table 2 provides an overview of all used quantization transforms, including our novel one for eight power moments. For trigonometric moments, we do not apply such quantization transforms. The Fourier basis makes good use of the available memory as is and the variants of our technique using trigonometric moments at 16 bits per real moment do not provide particularly interesting tradeoffs anyway.

When using one of these quantization transforms, the correct way to incorporate a new fragment at depth z_l with opacity α_l into the existing normalized, quantized vector of moments $\Theta_m^\star\left(\frac{b}{b_0}\right)$ is

$$\frac{b_0 \cdot \Theta_m^\star\left(\frac{b}{b_0}\right) - \ln(1 - \alpha_l) \cdot \Theta_m^\star(\boldsymbol{b}(z_l))}{b_0 - \ln(1 - \alpha_l)}.$$

3 ADDITIONAL RESULTS

In the following, we provide some additional results and timings for other variants of the discussed techniques.

3.1 Results With Eight Power Moments

In the paper, we do not show results of moment-based OIT with eight power moments because we believe that the variants with six

Authors' addresses: Cedrick Münstermann, s6bemuen@uni-bonn.de; Stefan Krumpen, krumpen@cs.uni-bonn.de; Reinhard Klein, rk@cs.uni-bonn.de, University of Bonn, Bonn, 53115, Endenicher Allee 19a, Germany; Christoph Peters, christoph.peters@kit.edu, Karlsruhe Institute of Technology, Karlsruhe, 76131, Am Fasanengarten 5, Germany, University of Bonn, Germany.

2577-6193/2018/5-ART7 $15.00
https://doi.org/10.1145/3203206

Moments	Quantization	Vector of biasing moments $b^\star$	Moment bias α
4 power moments	80 bits	$(1, 0, 0.628, 0, 0.628)^\mathsf{T}$	$6 \cdot 10^{-5}$
4 power moments	160 bits	$(1, 0, 0.375, 0, 0.375)^\mathsf{T}$	$5 \cdot 10^{-7}$
6 power moments	112 bits	$(1, 0, 0.5566, 0, 0.489, 0, 0.47869382)^\mathsf{T}$	$6 \cdot 10^{-4}$
6 power moments	224 bits	$(1, 0, 0.48, 0, 0.451, 0, 0.45)^\mathsf{T}$	$5 \cdot 10^{-6}$
8 power moments	144 bits	$(1, 0, 0.424749164, 0, 0.224078027, 0, 0.153692308, 0, 0.129004405)^\mathsf{T}$	$2.5 \cdot 10^{-3}$
8 power moments	288 bits	$(1, 0, 0.75, 0, 0.676666667, 0, 0.63, 0, 0.60030303)^\mathsf{T}$	$5 \cdot 10^{-5}$
2 trigonometric moments	80 bits	$(1, 0, 0)^\mathsf{T}$	$4 \cdot 10^{-4}$
2 trigonometric moments	160 bits	$(1, 0, 0)^\mathsf{T}$	$4 \cdot 10^{-7}$
3 trigonometric moments	112 bits	$(1, 0, 0, 0)^\mathsf{T}$	$6.5 \cdot 10^{-4}$
3 trigonometric moments	224 bits	$(1, 0, 0, 0)^\mathsf{T}$	$8 \cdot 10^{-7}$
4 trigonometric moments	144 bits	$(1, 0, 0, 0, 0)^\mathsf{T}$	$8.5 \cdot 10^{-4}$
4 trigonometric moments	288 bits	$(1, 0, 0, 0, 0)^\mathsf{T}$	$1.5 \cdot 10^{-6}$

Table 1. The optimized vector of biasing moments and the recommended moment bias for all variants of moment-based OIT.

$$\Theta_4^\star \begin{pmatrix} b_1 \\ b_2 \\ b_3 \\ b_4 \end{pmatrix} := \begin{pmatrix} \frac{3}{2} & 0 & -2 & 0 \\ 0 & 4 & 0 & -4 \\ \frac{1}{2} \cdot \sqrt{3} & 0 & -\frac{2}{9} \cdot \sqrt{3} & 0 \\ 0 & \frac{1}{2} & 0 & \frac{1}{2} \end{pmatrix} \cdot \begin{pmatrix} b_1 \\ b_2 \\ b_3 \\ b_4 \end{pmatrix} + \begin{pmatrix} 0.5 \\ 0 \\ 0.5 \\ 0 \end{pmatrix}$$

$$\Theta_6^\star \begin{pmatrix} b_1 \\ b_2 \\ b_3 \\ b_4 \\ b_5 \\ b_6 \end{pmatrix} := \begin{pmatrix} \begin{pmatrix} 2.5 & -10 & 8 \\ -1.87499864 & 4.20757543 & -1.83257679 \\ 1.26583039 & -1.47644883 & 0.71061660 \end{pmatrix} \cdot \begin{pmatrix} b_1 \\ b_3 \\ b_5 \end{pmatrix} + \begin{pmatrix} 0.5 \\ 0.5 \\ 0.5 \end{pmatrix} \\ \begin{pmatrix} 4 & -4 & 0 \\ 9 & -24 & 16 \\ -0.57759806 & 4.61936648 & -3.07953907 \end{pmatrix} \cdot \begin{pmatrix} b_2 \\ b_4 \\ b_6 \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ 0.018888946 \end{pmatrix} \end{pmatrix}$$

$$\Theta_8^\star \begin{pmatrix} b_1 \\ b_2 \\ b_3 \\ b_4 \\ b_5 \\ b_6 \\ b_7 \\ b_8 \end{pmatrix} := \begin{pmatrix} \begin{pmatrix} 3.48044636 & -27.57607375 & 55.12673843 & -31.53111104 \\ 1.26797186 & -0.92875581 & -2.07520453 & 1.23598848 \\ -2.16715600 & 6.17950200 & -0.27651557 & -4.23583042 \\ 0.97433288 & -0.44342683 & -0.36049165 & 0.31014947 \end{pmatrix} \cdot \begin{pmatrix} b_1 \\ b_3 \\ b_5 \\ b_7 \end{pmatrix} + \begin{pmatrix} 0.5 \\ 0.5 \\ 0.5 \\ 0.5 \end{pmatrix} \\ \begin{pmatrix} 0.28050413 & -0.75763384 & 0.39217959 & -0.88753187 \\ -2.01362266 & 0.22155137 & -1.06107954 & 2.83887202 \\ -7.31010495 & 13.98559797 & -0.11430577 & -7.43618994 \\ -15.89542156 & 79.61863271 & -127.45727899 & 63.73494567 \end{pmatrix} \cdot \begin{pmatrix} b_2 \\ b_4 \\ b_6 \\ b_8 \end{pmatrix} + \begin{pmatrix} 0.97248199 \\ 1 \\ 0.99917919 \\ 0.99177829 \end{pmatrix} \end{pmatrix}$$

Table 2. The quantization transforms to be used when storing four, six or eight power moments in 16 bits per moment.

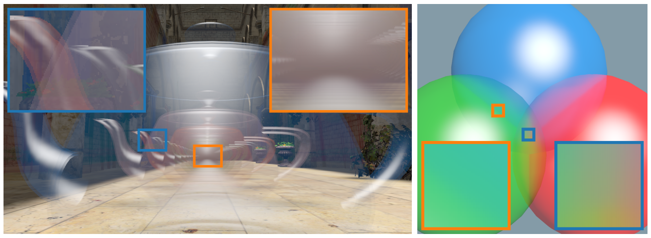
power moments or three trigonometric moments are more useful. As shown in Figure 2, eight power moments provide a slight quality improvement over six power moments but are noticeably worse than three trigonometric moments. From Table 3 we find that the run time cost of eight power moments is closer to three trigonometric moments. Thus, it does fall into the gap between these techniques in terms of cost and quality but the quality improvement is relatively small compared to the increased cost. Another drawback is the inability to deal with intersections (see Figure 1).

3.2 Efficiency of Rasterizer Ordered Views

Table 3 also sheds some light on an interesting and unexpected implementation detail. In an effort to determine the overhead of using rasterizer ordered views for moment-based OIT, we implemented the variants using single-precision floats in this manner. Surprisingly, the resulting frame times on our test hardware are actually shorter than those obtained with hardware-accelerated additive blending. We do not have a conclusive explanation for this phenomenon but note that a low overhead from using rasterizer ordered views is expected since the shader program is very short.

3.3 Multi-Layer Alpha Blending with Low Dynamic Range

Like most modern assets, the assets used in our evaluation require rendering with more than 8 bits per color channel to reproduce their appearance adequately. Therefore, we have used render targets with 16 bits per color channel consistently for all techniques. However, multi-layer alpha blending was originally proposed as a technique using low dynamic range and thus our implementation is slower than the original technique. Table 3 includes timings for a variant of multi-layer alpha blending that uses only 8 bits per color channel and for the transmittance. Note that this variant does not create correct images for our scenes.



(a) Teapots

(b) Intersecting spheres

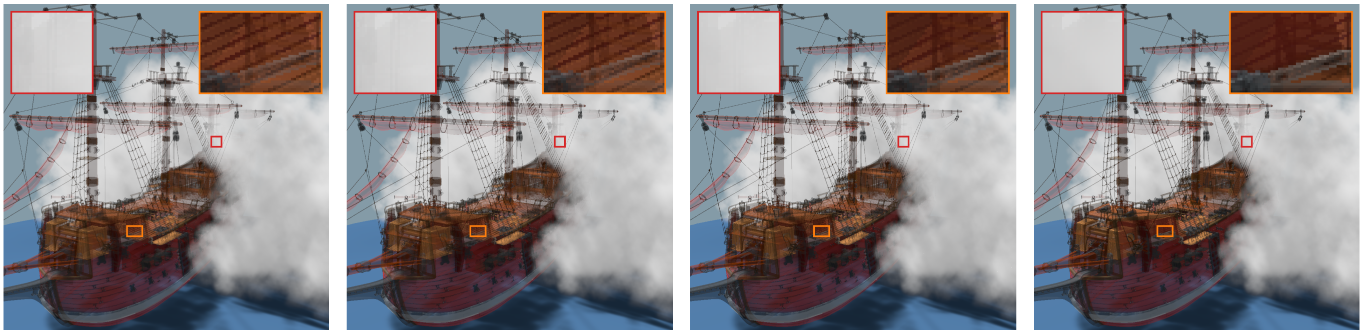
Fig. 1. Results for further scenes using eight power moments stored in 288 bits.

The speedup achieved by reducing the bandwidth requirements in this manner is significant. Multi-layer alpha blending with four layers now has a run time close to that of our technique with six power moments stored in 224 bits. Still, our technique offers better quality and handles the high-dynamic range rendering with substantially lower overhead.

REFERENCES

Christoph Peters and Reinhard Klein. 2015. Moment Shadow Mapping. In *Proceedings of the 19th ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games (i3D '15)*. ACM, 7–14. https://doi.org/10.1145/2699276.2699277

Christoph Peters, Cedrick Münstermann, Nico Wetzstein, and Reinhard Klein. 2017. Improved Moment Shadow Maps for Translucent Occluders, Soft Shadows and Single Scattering. *Journal of Computer Graphics Techniques (JCGT)* 6, 1 (2017), 17–67. http://jcgt.org/published/0006/01/03/



(a) Moment-based OIT, ours, 6 power moments, 224 bits, 14 ms

(b) Moment-based OIT, ours, 8 power moments, 288 bits, 16 ms

(c) Moment-based OIT, ours, 3 trigonometric moments, 224 bits, 16 ms

(d) Ground truth, depth peeling, 123 ms

Fig. 2. Results for the large ship scene from the paper with further variants of our technique.

OIT technique		Adaptive transparency							Multi-layer alpha blending											
Nodes/layers		2	3	4	5	6	7	8	1		2		3		4		5		6	
Memory		64	96	128	160	192	224	256	48	80	96	160	144	240	192	320	240	400	288	480
Teapots	Pass 1	1.0	1.3	1.6	1.9	2.2	2.5	2.8	1.4	1.5	1.8	2.0	2.3	2.7	2.8	3.4	3.3	4.1	3.9	4.8
	Pass 2	0.88	0.90	0.94	1.0	1.1	1.2	1.3	0.07	0.09	0.09	0.13	0.11	0.17	0.13	0.22	0.16	0.26	0.18	0.30
	Diff.	1.1	1.4	1.8	2.1	2.5	3.0	3.4	0.69	0.78	1.1	1.4	1.6	2.0	2.1	2.8	2.6	3.5	3.2	4.3
Ship	Pass 1	1.9	2.5	3.0	3.4	4.0	4.6	5.4	2.8	3.0	3.5	3.9	4.4	5.2	5.5	6.6	6.5	8.5	7.9	10.1
	Pass 2	1.6	1.7	1.8	1.9	2.0	2.1	2.2	0.05	0.07	0.07	0.11	0.09	0.15	0.12	0.20	0.14	0.24	0.16	0.28
	Diff.	2.1	2.7	3.3	3.8	4.5	5.2	6.2	1.4	1.5	2.0	2.6	2.9	3.8	4.1	5.3	5.1	7.2	6.5	8.9

Moment type		Power									Trigonometric								
Moment count		4			6			8			2			3			4		
Rasterizer ordered view		yes		no	yes		no	yes		no	yes		no	yes		no	yes		no
Memory		80	160	160	112	224	224	144	288	288	80	160	160	112	224	224	144	288	288
Teapots	Pass 1	0.94	1.2	1.8	1.2	1.6	2.5	1.3	1.9	3.1	0.94	1.2	1.8	1.2	1.6	2.5	1.3	1.9	3.2
	Pass 2	1.0	1.0	0.94	1.1	1.2	1.1	1.4	1.4	1.3	1.2	1.2	1.1	1.7	1.7	1.6	2.4	2.4	2.3
	Diff.	1.2	1.5	2.0	1.6	2.1	2.8	2.0	2.6	3.7	1.4	1.7	2.2	2.2	2.6	3.4	3.0	3.6	4.7
Ship	Pass 1	2.0	2.3	3.3	2.3	3.1	4.5	2.6	3.5	5.5	1.9	2.3	3.3	2.3	3.1	4.8	2.9	3.6	5.9
	Pass 2	1.8	1.8	1.8	2.2	2.2	2.1	2.7	2.6	2.5	2.2	2.2	2.2	3.2	3.2	3.1	4.5	4.8	4.3
	Diff.	2.3	2.7	3.6	3.1	3.8	5.1	3.9	4.7	6.8	2.7	3.1	4.0	4.2	4.9	6.4	5.7	6.8	8.7

Table 3. An extended version of the table of frame times in the paper. Differential timings are full frame times where we have subtracted the timings for rendering with simple alpha blending (1.9 ms for the teapots, 2.7 ms for the ship). Variants of multi-layer alpha blending with less memory use 8 bits per color channel.