Supplemental material:
Primary-Space Adaptive Control Variates using Piecewise-Polynomial Approximations

MIGUEL CRESPO, Universidad de Zaragoza - I3A
ADRIAN JARABO, Universidad de Zaragoza - I3A, and Centro Universitario de la Defensa Zaragoza
ADOLFO MUÑOZ, Universidad de Zaragoza - I3A

This is the supplemental material of the paper *"Primary-Space Adaptive Control Variates using Piecewise-Polynomial Approximations"* submitted to ACM Transactions on Graphics. It includes:

- **Section S.1** - Detailed convergence analysis.
- **Section S.2** - Additional analysis of our adaptive residual ratio tracking.
- **Section S.3** - Breakdown of the temporal and memory cost of our technique.
- **Section S.4** - Comparison between our unbiased technique and pure adaptive nested quadrature integration.
- **Section S.5** - Results of combining our technique with denoising.
- **Section S.6** - Analysis of the influence of the variance of higher dimensions during the construction of the piecewise polynomial.
- **Section S.7** - Analysis of our method's convergence when the control variate approximates poorly the signal.

S.1 DETAILED CONVERGENCE ANALYSIS

In this appendix we show the individual integrals used to analyze the performance of our technique (Section 4.4 in the main document) depending on the number of samples used to build the control variate and for estimating the residual. Figures 1 (2D), 2 (3D) and 3 (4D) show the individual integrals including their grayscale representation along with the corresponding control variate approximation and the residual. Also, as in Figure 4 in the main document, we present for each integral the error, time and the product between cost and error. For each function, we also include the performance when integrating the full domain and projecting the integrand into buckets

We start by analyzing the lower dimensional experiments, featured in Figure 1 where we use our technique to integrate 2D functions. It can be seen how the performance is directly correlated with the high-frequency details of the functions: functions like the one in the first row, which is low frequency, are easily integrated (as can be seen in the control variate and residual). However, function with more complex details (like the last two rows) generate a higher residual because the control variate has more trouble fitting such high frequency details. Nevertheless, the benefits of our technique improve when projecting the integral into a set of buckets due to the reusing of the control variate between all bins.

Authors' addresses: Miguel Crespo, mcrescas@gmail.com, Universidad de Zaragoza - I3A; Adrian Jarabo, ajarabo@unizar.es, Universidad de Zaragoza - I3A, and Centro Universitario de la Defensa Zaragoza; Adolfo Muñoz, adolfo@unizar.es, Universidad de Zaragoza - I3A.

In Figure 2 where we analyze 3D functions. The left column show a set of different 2D slices of the function. It can be seen how the increased dimensionality penalizes the efficiency of the control variate samples (quadrature, horizontal axis) while Monte Carlo samples for the residual maintain its performance, but as in the previous experiments, projecting into buckets improves the overall performance. This trend is also followed by 4D functions (Figure 3, left column shows a subset of the 2D slices of the function), where the efficiency of the quadrature samples is reduced even further but again the projection into buckets makes it worth the performance decrease.

Furthermore, note that as the nested quadrature technique (bottom row in all performance plots) is deterministic, the convergence along its number of samples is not predictably decreasing: in some circumstances, adding new subdivisions might actually increase the error, and some error minima appear for a specific integrand and a specific number of samples. The global tendency is, nevertheless, decreasing error as the number of samples increase. Finally, the global tendencies identified in Section 4.4 of the paper also hold in individual function calculations with small per-function variations.

S.2 ADDITIONAL ANALYSIS OF ADAPTIVE RESIDUAL RATIO TRACKING

In this section we present more convergence analysis of our technique used to compute transmittance in heterogeneous media, which was presented in Section 5 of the main document. Figure 4 shows the convergence of our method compared to residual ratio tracking [Novák et al. 2014] with constant precalculated μ_c (set to $\mu_c = \int_0^t \mu(\mathbf{x}_s)ds$, which is the optimal parameter according to the authors) and delta tracking [Woodcock et al. 1965] for three different procedural heterogeneous media. In all cases we use the same tight majorant $\bar{\mu} = \max_{\mathbf{x}}(\mu(\mathbf{x}))$. We build our polynomial approximation performing three iterations, so that the introduced overhead is not very dramatic. These three iterations result into nine additional queries to the medium extinction.

As expected, the performance of both residual ratio tracking [Novák et al. 2014] and our method relate with the quality of approximation (i.e., the variance in the residual). For the cases where residual ratio tracking performs well (as in medium a) our technique in general performs similarly, since the approximation does not introduce significant error compared with a constant control. However, in cases where our technique is able to faithfully adapt to the signal (e.g., in cases with non-uniform densities like in media b and c), it significantly outperforms residual ratio tracking, without introducing a significant overhead (as can be seen in the light blue plots showing

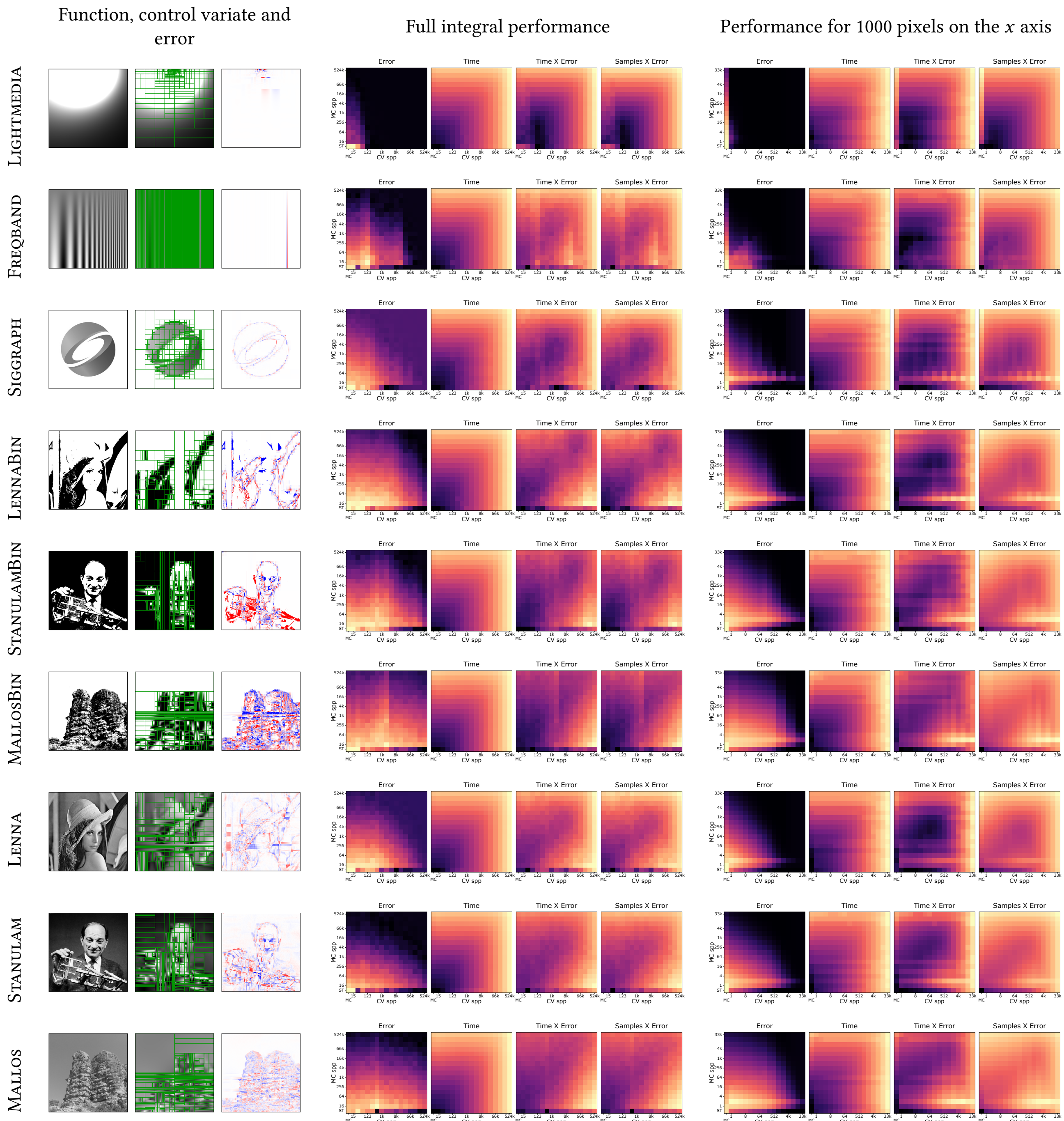

Fig. 1. Integration of a two-dimensional function (image) per row. From left to right: grayscale representation of the function; piecewise polynomial control variate (boundaries of each region marked in green); residual of the control variate (underestimation in red, overestimation in blue); two dimensional convergence plots (log scale) exploring samples per pixel for a full 2D integral; two dimensional convergence plots (log scale) with amortization of control variate samples along 1000 buckets (pixels) in the horizontal axis. For each convergence plot, we show error, time, the product of both and the product of total number of samples and error in a colored logarithmic scale.

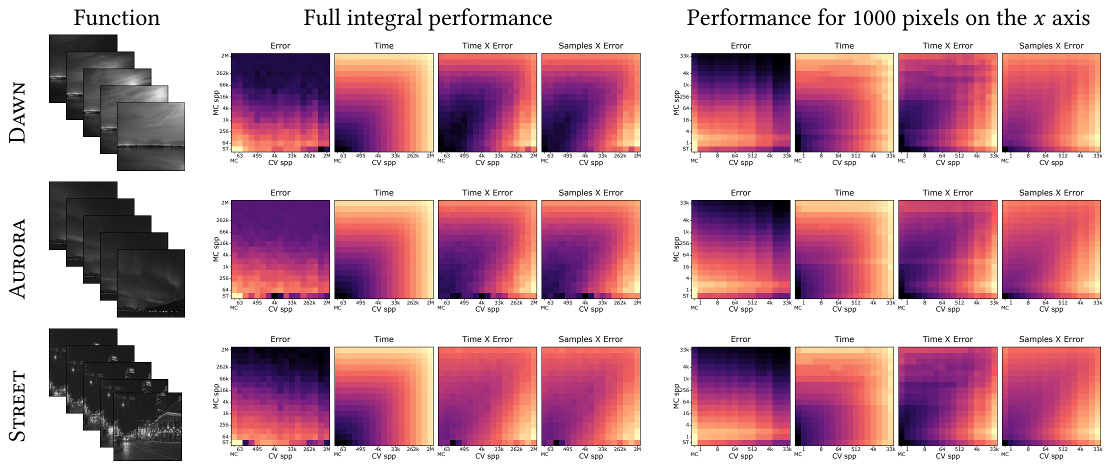

Fig. 2. Integration of a three-dimensional function (video) per row. From left to right: grayscale representation of the function consisting in different frames of the sequence; two dimensional convergence plots (log scale) exploring samples per pixel for a full 3D integral; two dimensional convergence plots (log scale) with amortization of control variate samples along 1000 buckets (pixels) in the horizontal axis. For each convergence plot, we show error, time, the product of both and the product of total number of samples and error in a colored logarithmic scale.

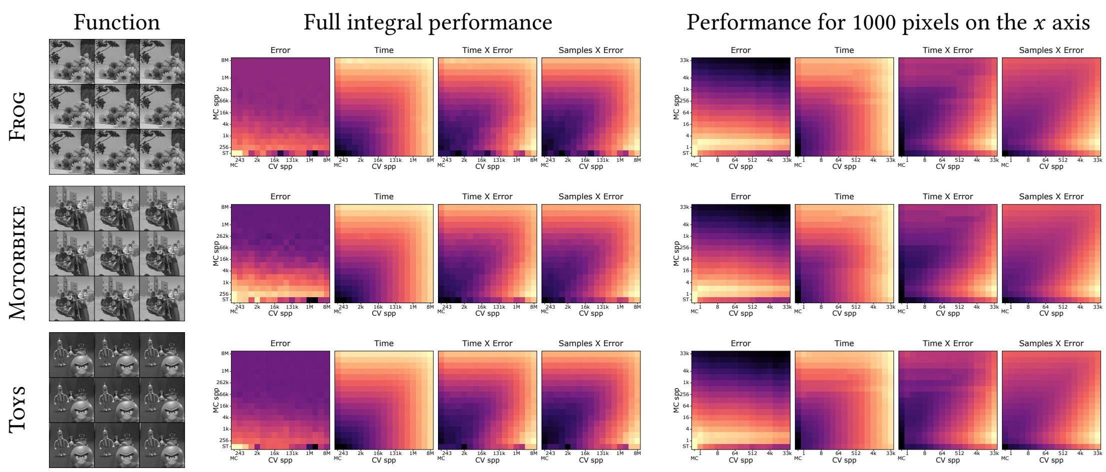

Fig. 3. Integration of a four-dimensional function (lightfield) per row. From left to right: grayscale representation of the function consisting in a subset of the images constituting the lightfield; two dimensional convergence plots (log scale) exploring samples per pixel for a full 4D integral; two dimensional convergence plots (log scale) with amortization of control variate samples along 1000 buckets (pixels) in the horizontal axis. For each convergence plot, we show error, time, the product of both and the product of total number of samples and error in a colored logarithmic scale.

the residual values taken respectively from the control extinction $\mu_c(x_s)$).

S.3 TEMPORAL AND MEMORY COST OF OUR TECHNIQUE

In this section we present the temporal breakdown of our technique separated into each stage. We illustrate the temporal distribution for the scenes of the paper in Table 1. This distribution is affected by the number of dimensions of the control variate. For lower dimensionalities the overhead is smaller compared to the cost of evaluating the integrand (12% to 34%) but higher dimensionality increases such overhead (up to a 75.5% with 6 dimensions). Also, the temporal breakdown is greatly affected with the cost of evaluating the integrand: in very simple scenes such cost is small so the overhead of our integration techniques is more noticeable, while in the case of very complex scenes (many geometrical primitives) our overhead can become negligible (which is coherent with the convergence tests in the paper). Between different stages, generating the control variate has usually a higher cost than evaluating it. However, the higher overhead generally comes from integrating the control variate.

Furthermore, we show in Table 2 a comparison of the memory usage of the scenes presented in Section 8 in the main document, in

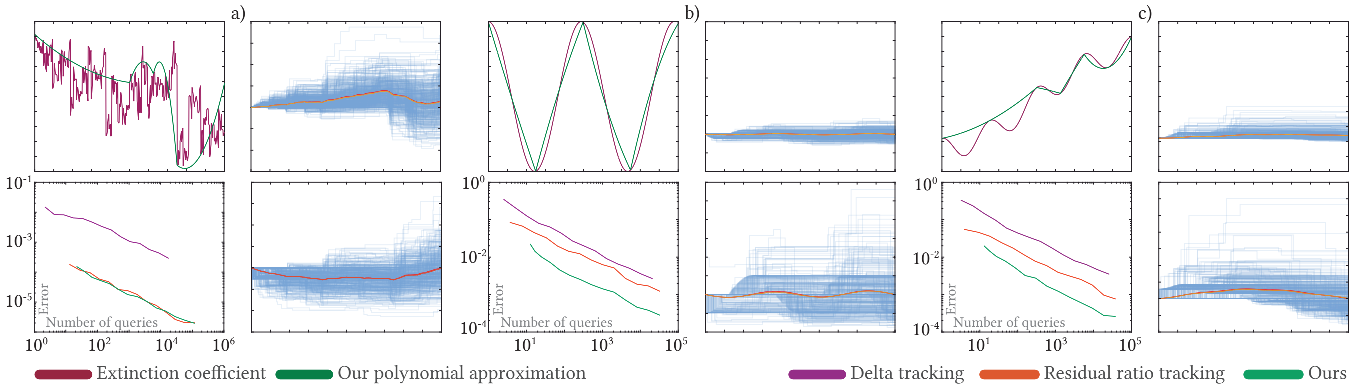

Fig. 4. Comparison of our adaptive residual ratio tracking against delta tracking [Woodcock et al. 1965] and residual ratio tracking with constant control [Novák et al. 2014], for three different procedural heterogeneous media. The extinction of each medium is generated using a) fractional Brownian noise, b) a cosine function, and c) a cosine function weighted by a wide sigmoid. For each medium we show (left to right): (top) The extinction coefficient of the medium (garnet) and our polynomial control approximation (dark green); (bottom) the convergence of delta tracking (purple), residual ratio tracking (orange), and our adaptive residual ratio tracking (green); the residual and numerical approximation of our method (top) and Novak's residual ratio tracking (bottom), both for 512 samples (shown in light blue).

which we show a reasonable but manageable overhead over Monte Carlo in terms of memory consumption that is used for storing the control variate. Compared to Hachisuka et al.'s method [2008] (which also store samples in a specific data structure) our technique presents a much lower memory footprint. This is due to the fact Hachisuka et al.'s method requires to store all samples in a K-D tree data structure while our algorithm requires to store only the samples for the control variate (1/16 of the total samples)

S.4 COMPARISON WITH NESTED QUADRATURE

In this section we compare our approach with pure adaptive nested quadrature rules for several scenes. Note that we have already included in Section 6 of the main text a comparison between quadrature integration [Muñoz 2014] in the context of low-order scattering, but we extend such analysis for higher dimensionality. This is equivalent to rendering with our control variate without dedicating any Monte Carlo sample for the residual.

We analyze Simpson-Trapezoidal (orders two and one, as our control variate) and Boole-Simpson (orders four and two) and compare them with our results for the same number of samples.

In Figure 5 we show a comparison between the adaptive nested quadrature introduced above versus our Adaptive Residual Ratio Tracking (Section 5 of the main text) at an equal number of queries to the media. Note that using pure quadrature for estimating the transmittance, even though it is a soft domain, smaller fine-grained details are missed. The resulting artifacts can be seen in Smoke scene, where Simpson-Trapezoidal rule miss a lot of details even when applied adaptively. On the other hand, Boole-Simpson nested rule reduces the number of artifacts but does not eliminate them. In contrast, our technique keeps the benefits of the low-frequency domain but recovering all high-frequency details of the heterogenous medium.

Figure 6 shows an equal time comparison of quadrature rules versus our technique for rendering distribution effects (Section 8

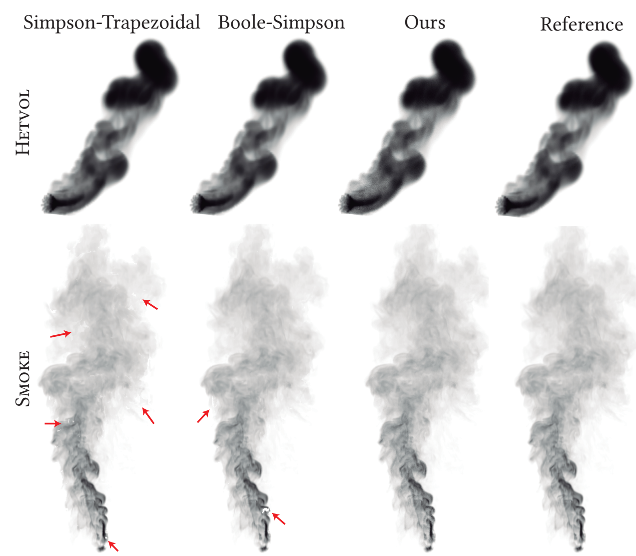

Fig. 5. Comparison between adaptive nested quadrature using Simpson-Trapezoidal rules (left) and Boole-Simpsons rules (middle) versus our Adaptive Residual Ratio Tracking (right) at an equal number of queries to the media. Artifacts in the images has been marked with a red arrow for easier viewing. Note how depending on the heterogeneity of the media, quadrature rules can miss portion of the signal ignoring fine-grained details. In contrast, our technique is able to recover from that misses.

of the main text). Again, quadrature rules have trouble integrating heavy gradients such as occlusion boundaries in the scene, and some important high frequency features are even missed (see Chess and Helicopter scenes for example). By incrementing the order of the quadrature rule (Bool-Simpson) more details are reconstructed (boundaries are improved) but the Runge phenomenon appears:

Table 1. We present in this table a temporal breakdown of our technique (in percentages) of the time to generate the regions of the control variate, the time it takes to compute the residual and the time needed to do the sampling. We also show the temporal cost of both evaluating and integrating the control variate plus the time that cost to generate the data structure used in our implementation (all already included in the residual time). We don't show time for obtaining the optimal alpha because it is negligible. Last column indicate the dimensionality of the control variate and (*) mark that although it is built using only that dimensions, the integral has more dimensions.

Scene	Generate CV	Residual	Sampling	Evaluate CV	Integrate CV	Data structure	Dim CV
Pumpkin	3.1%	9.11%	87.7%	0.01%	0.5%	8.6%	3D
Occluder	6.9%	81.73%	11.3%	0.07%	9.2%	72.46%	3D
Laser	1%	97.62%	1.33%	0.02%	41%	56.6%	4D
GreenDragon	1.1%	83.1%	15.76%	0.03%	35.87%	47.2%	4D
Pool	6%	27%	66%	0.09%	2.1%	24.81%	3D
Chess	6%	20%	73%	0.05%	5%	14.95%	4D
Helicopter	3.1%	64%	31%	19.7%	28.7%	15.6%	5D
Volley Balls	1.8%	73.7%	24%	48%	21%	4.7%	6D
Violin	2.7%	71%	25%	0.01%	51%	19.99%	4D
House	4.57%	61%	33.55%	0.13%	25.49%	35.38%	4D
Classroom	3.38%	46%	50.5%	0.14%	21.27%	24.59%	4D
Bistro	1.09%	6.81%	92.57%	0.083%	2.8%	3.97%	4D*

Table 2. Report of the memory usage (in GiB) of the results shown in Section 8 in the main document. Monte Carlo can be interpreted as the baseline of the memory consumed by the geometry of the scene. Note that, although our method has a higher memory consumption than Monte Carlo, it still remains manageable as the number of samples increases.

Scene	Spp	Resolution	Monte Carlo	[Hachisuka et al. 2008]	Ours
Pool	64	875x1000	0.100	31.959	0.255
Chess	64	750x1000	0.304	29.607	0.497
Helicopter	64	960x540	0.184	27.489	0.532
Volley Balls	64	720x480	0.312	23.170	0.478

unwanted oscillations of the polynomial near boundaries appear as perceivable artifacts.

In contrast, our technique does not suffer from these artifacts achieving sharp high-frequency details and recovering the signal that could be missed in the control variate. This same behavior can be seen in Figure 7 where we compare the same techniques in the domain of direct light integration (Section 7 of the main text).

Generally speaking, quadrature rules are biased, while our approach is not, and such bias results in unpleasant structured artifacts that can be perceivable. Besides our faster convergence, the error from our technique is not structured but high frequency, which is visually more pleasant.

Fig. 6. Equal time comparison of integrating distribution effects between using adaptive nested quadrature Simpson-Trapezoidal (left) and Boole-Simpson (middle) rules versus our technique (right). Note how quadrature rules has trouble in high-frequency edges of the scene and when the dimensionality increase (from top to bottom: 3D, 4D, 5D, 6D). However, our technique is capable of recover both low-frequency areas and high-frequency details of the scenes.

S.5 COMBINATION WITH DENOISING

Our technique combines the advantages of polynomial approximations without incurring in bias (by including stochastic evaluation using Monte Carlo integration of the residual). When we calculated our residual, we distribute our samples to correct the "existing" bias of the piecewise polynomial approximation, generating high frequency noise. A popular technique that can improve this (when adding more sample budget can be prohibitive) is denoising the results.

State of the arts denoisers are capable of achieve noise-free images generated with a low-sampling regime. Given that the residual only has that high-frequency noise, we can apply a denoiser to remove it. We use Bako et al.'s work [2017], which needs several buffers with information about the image to be denoised: first it need the radiance separated in its diffuse and specular components. Secondly, it needs some auxiliary information about the scene as normals, albedo and depth (we gather it using the residual's samples information). Finally,

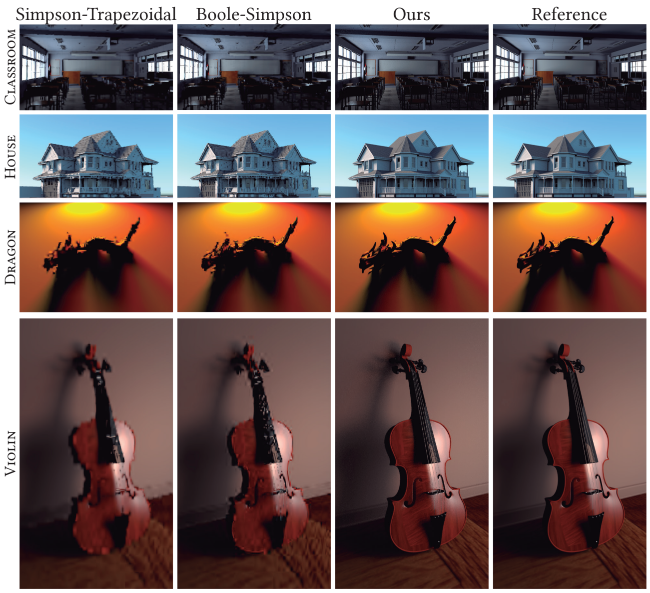

Fig. 7. Comparison between using adaptive nested quadrature Simpson-Trapezoidal (left) and Boole-Simpson (middle) rules versus our technique (right) at an equal time for integrating direct illumination. Note how quadrature rules keep clean the low-frequency areas but suffers from achieving sharp edges (even adding artifacts). Moreover, our technique is able to recover the fine-grained details in the borders while working smoothly in soft shadows.

for all the above information, its variance per pixel is also calculated (we use the variance of the residual estimation).

Figure 8 shows some denoising experiments in images generated by our technique using only 8 samples per pixel (the minimum amount usually computed in combination with denoisers). In such low-sampling regime, our technique does not excels versus Monte Carlo, even introducing noise that cannot be processed well using denoisers. On the other hand, Figure 9 shows results using a higher sampling count, in which our method can be processed correctly by denoisers, but introducing bias as blurred regions in the images (i.e., shadows in the bench or borders in Volley scene). In conclusion, our technique is not only agnostic to importance sampling strategies, but also to post-processing denoising algorithms. Being unbiased, our technique clearly benefits from the usage of a denoiser.

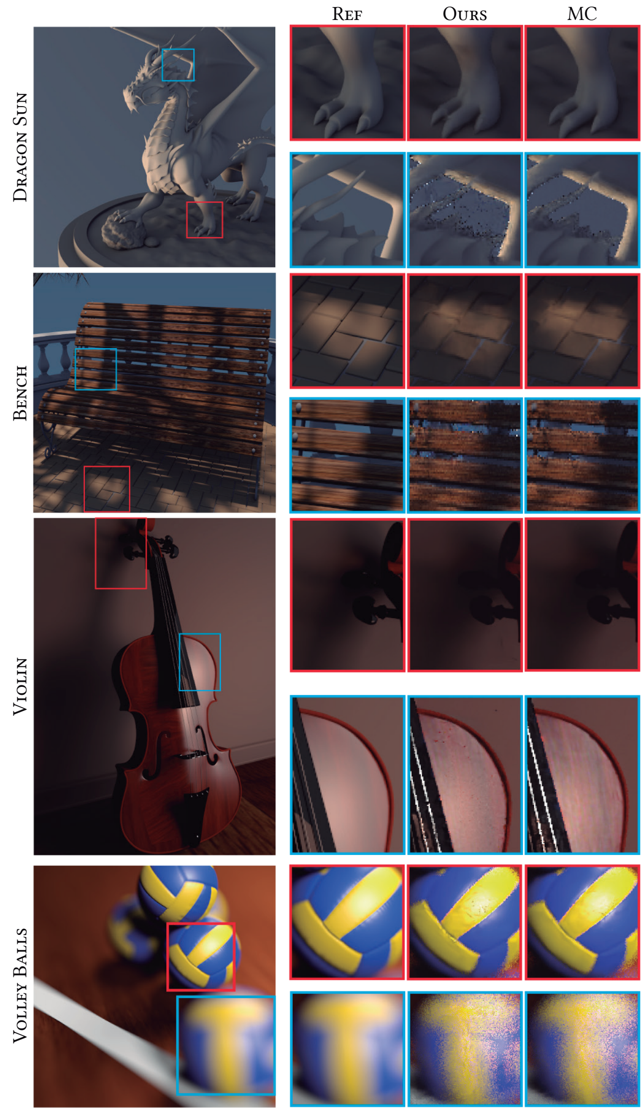

Fig. 8. Comparison of denoised results from both our technique and Monte Carlo (left image is the reference). Owing to the number of samples is very low (8spp), our technique is not able to obtain a good control variate and shows more visible noise.

S.6 VARIANCE IN HIGHER-ORDER QUADRATURE

In Section 9 of the main text we analyze the performance of our control variate in high-dimensional problems by building a low-dimensional control variate on top of an estimate of the high-dimensional integral. However, this strategy introduces variance which depends on the number of samples used in the estimator.

To analyze the effect of such variance in the result, we present in Figure 10 an experiment that we compute using different number of samples to estimate the higher dimensions. Note that we are not applying the residual correction, therefore we have only the piecewise polynomial with artifacts in discontinuities. It can be seen how the number of Monte Carlo samples used the construction of the control variate affect directly the quality of the illumination captured in the approximation: Using only one sample lead to noticeable artifacts. On the other hand, we observe that increasing sampling count beyond four Monte Carlo samples do not improve significantly the accuracy of the final result.

S.7 ANALYSIS OF CONVERGENCE IN ARTIFACTS

In this section we analyze how evolve the convergence of our method when the control variates fails to adapt to the signal. This effect can be seen as rectangular zones of the image with an increase

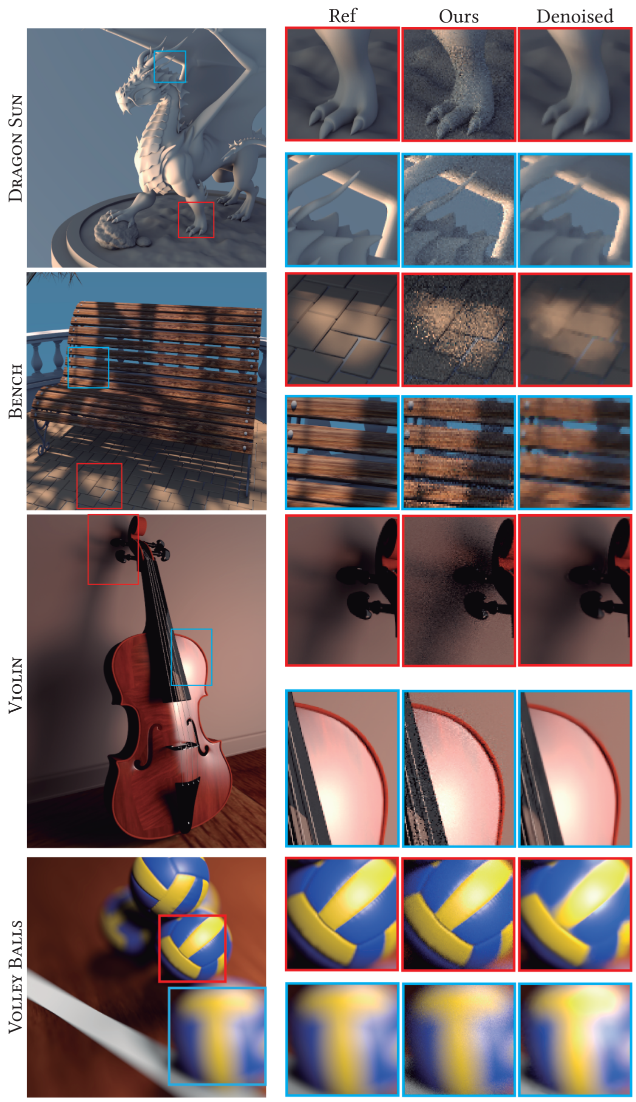

Fig. 9. Comparison between our results versus those obtained after denoising (left image is the reference). It can be seen that our technique can be denoised using conventional techniques while having the same problems related with overblurring or artifacts coming from the biased denoiser.

of noise respecting to the near areas, being something similar to what happen with other adaptive methods.

Figure 11 shows 3 different insets of regions dominated with such artifacts with an increasing number of samples. While our method suffers from that challenge with low sampling counts, it can be seen how such artifacts disappear at the same rate as the control variate improves. Because of our ratio explained in Section 4.4 in the main document, the refinement of the adaptive approximation is linked directly with the computation budget, therefore our method always improves. In addition, we present in Section 4.2 in the main document that our error estimation takes into account the size of the divisions, which allows to refine the areas where the artifacts

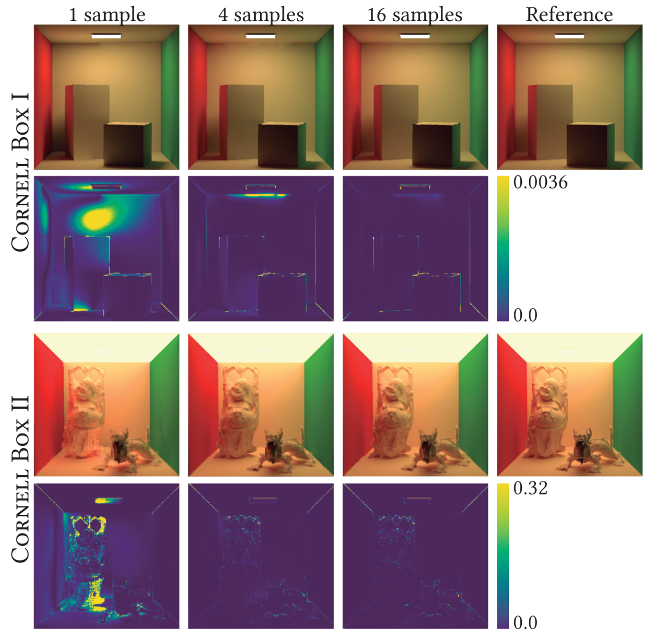

Fig. 10. Experiment of how the number of samples used to estimate the high-dimensional integral influence the piecewise approximation. For each rendering we show in addition the error map with respect to the reference. Note how a few samples are enough to avoid the most noticiable light artifacts. All results are calculated using 256 spp.

are generated because of a erroneous estimation of the error. Nevertheless, our method's error is always lower than Monte Carlo integration even in that regions.

REFERENCES

Steve Bako, Thijs Vogels, Brian McWilliams, Mark Meyer, Jan Novák, Alex Harvill, Pradeep Sen, Tony Derose, and Fabrice Rousselle. 2017. Kernel-predicting Convolutional Networks for Denoising Monte Carlo Renderings. *ACM Transactions on Graphics* 36, 4 (2017), 97.

Toshiya Hachisuka, Wojciech Jarosz, Richard Peter Weistroffer, Kevin Dale, Greg Humphreys, Matthias Zwicker, and Henrik Wann Jensen. 2008. Multidimensional Adaptive Sampling and Reconstruction for Ray Tracing. In *ACM Transactions on Graphics*, Vol. 27. ACM, 33.

Adolfo Muñoz. 2014. Higher Order Ray Marching. In *Computer Graphics Forum*, Vol. 33. Wiley Online Library, 167–176.

Jan Novák, Andrew Selle, and Wojciech Jarosz. 2014. Residual Ratio Tracking for Estimating Attenuation in Participating Media. *ACM Transactions on Graphics* 33, 6 (2014).

E. Woodcock, T. Murphi, P. Hemmings, and S. Longworth. 1965. Techniques Used in the GEM Code for Monte Carlo Neutronics Calculations in Reactors and Other Systems of Complex Geometry. In *Proceeding Conference Applications of Computing Methods to Reactors, ANL-7050.*

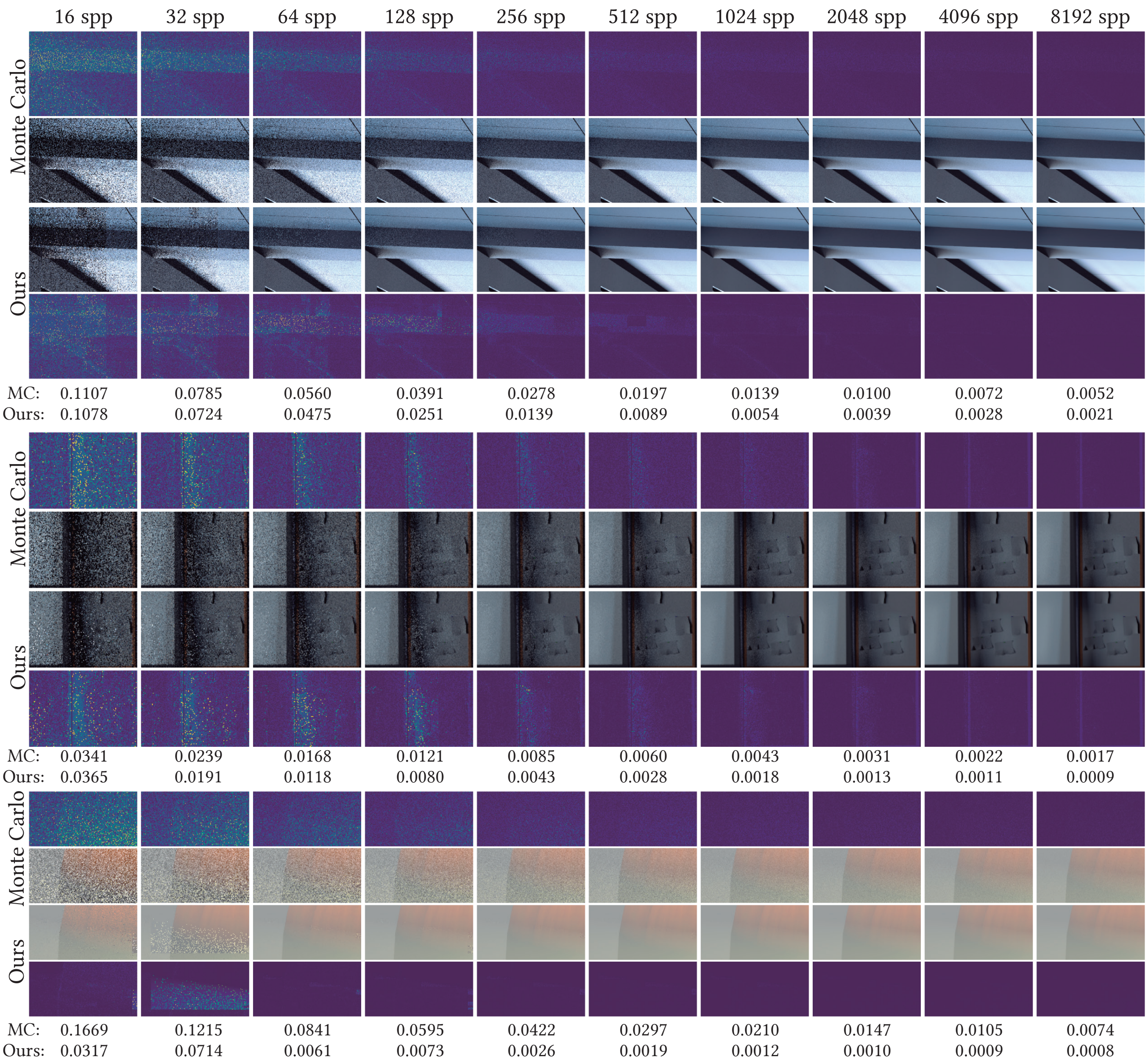

Fig. 11. Experiment comparing the convergence of our method versus Monte Carlo in areas of the image featuring a bad approximation of our control variate (at low sample count). First two rows are Classroom scene, and the last one is Pumpkin scene. Error metric is computed only in the inset area. Note how increasing the sample count is linked directly with how much refinement is done by our control variate and therefore, increasing it allows to remove any possible artifact in the final image.