A Hierarchical Automatic Stopping Condition for Monte Carlo Global Illumination

Holger Dammertz
Ulm University
James-Franck-Ring
Germany, 89081 Ulm

Johannes Hanika
Ulm University
James-Franck-Ring
Germany, 89081 Ulm

Alexander Keller
mental images GmbH
Fasanenstrasse 81
Germany, 10623 Berlin

Hendrik P.A. Lensch
Ulm University
James-Franck-Ring
Germany, 89081 Ulm

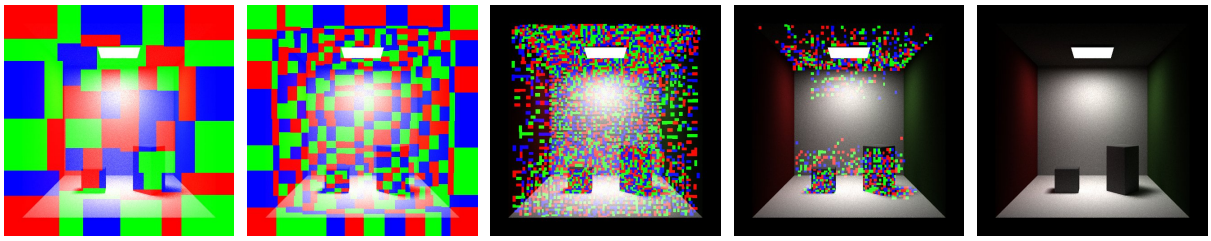

Figure 1: An example sequence of blocks. Initially large blocks (shown in saturated colors) are recursively refined and terminated individually as the image converges. Note that complex areas in the image, where indirect illumination dominates the appearance (e.g. the ceiling), are detected and refined by the algorithm.

ABSTRACT

We introduce a hierarchical image-space method to robustly terminate computations in Monte Carlo image synthesis, independent of image resolution. The technique consists of a robust convergence measure on blocks which are either recursively subdivided or terminated independently, using a criterion which separates signal and noise based on integral estimates from two separate sample sets. The technique can be easily implemented, as the evaluation of the error measure only requires a second framebuffer and a list of non-terminated blocks. Based on the stopping criterion, one can furthermore sample both the image plane as well as the light sources adaptively. Adaptive sampling reduces the number of samples required to gain the same root mean square error to one quarter in some of our test cases.

Keywords

Monte Carlo, Global Illumination

1 INTRODUCTION AND PREVIOUS WORK

Monte Carlo methods are among the most general techniques to solve the global illumination problem. Unbiased Monte Carlo methods are guaranteed to converge to the correct solution but the number of samples required for generating an image where the remaining noise is hardly visible is not known in advance.

We therefore propose a stopping algorithm which robustly separates the remaining noise in the rendered image from the variation due to the scene content, which is easy to implement, requires very little additional memory and we also show how to use it to adaptively sample from the light sources. Following the idea of Dippé and Wold [DW85], we estimate the rate of change as the difference between images generated with two different sample sets.

A lot of work has been done to quantify the perceived differences by the human visual system [Mys98, BM98, RPG99, YPG01, SCCD04, FP04, SGA*07, Dal93, MDMS05], error measures based on confidence intervals, contrast, variance, saliency and entropy have been investigated. For simplicity and fast evaluation, we base our stopping criterion on the remaining color noise in relation to the logarithmic luminance of the sample, but any other error measure could be used as well.

Rigau et al. [RFS03] presented a similar method based on entropy, but they only consider the image as a whole and sub-pixel refinements, not blocks of pixels.

Hachisuka et al. [HJW*08] also subdivide a hierarchy of samples very similar to the MISER algorithm [PTVF92], but our error measure is based

on two integral estimates, which is able to separate signal from noise. Also, additional reconstruction is not necessary because sampling is dense enough (at least one sample per pixel), and the samples do not have to be stored explicitly in our case.

In [ODR09] the authors describe an adaptive wavelet rendering that is similar to our approach but their scaling functions are more restrictive than our blocks and additionally we also adaptively sample the light source and are not restricted to 2d.

For a reliable error estimate, it is not sufficient to look at the variation inside a single pixel, due to the limited number of samples [Mit87]. To overcome this, we evaluate the error measure on a hierarchy of blocks, which are recursively refined only as the remaining error drops below a certain resolution-independent threshold (see Figure 1 for an illustration of the refinement process). This makes sure the algorithm adapts to the true signal, not a noisy estimate. Assuming that the samples available so far have not completely missed any major light contribution (i.e. one "firefly"-path is enough), our scheme operates conservatively.

Painter and Sloan [PS89] also used a hierarchy, an image-space *k*d-tree, but explicitly stored all samples and all levels of the hierarchy. We, on the other hand, simply store the integral estimates of the two sampling sets per pixel in two framebuffers. Furthermore, we only keep track of the leaf blocks, which significantly simplifies bookkeeping.

Based on our convergence estimate, adaptive sampling can be performed by simply supersampling only those blocks which are not yet terminated.

Note that this is different to most existing adaptive sampling approaches, since it is not based on placing samples in regions considered interesting at an early stage but splits and terminates rather conservatively. This is why the common problem of missed features such as small objects is not a problem.

Furthermore, as the technique is framebuffer-based, it is completely independent of the underlying rendering algorithm.

2 A HIERARCHICAL CRITERION

To make sure the algorithm does not stop computations before all important features have been detected, a lot of samples should be drawn before making a decision (see Figure 2 for an illustration, and Figure 3 for a comparison to a pure per-pixel approach). Since it is not desirable to shoot a lot of possibly unnecessary rays per pixel, we initially base the evaluation of the error on the image as a whole.

As the variance in the image will be distributed unevenly, depending on the problem and the type of path space sampler used, the stopping decision should be done locally. This is reflected by splitting the blocks when the algorithm has enough confidence not to have missed any important features, i.e. the per-block error measure drops below a certain threshold ε_s.

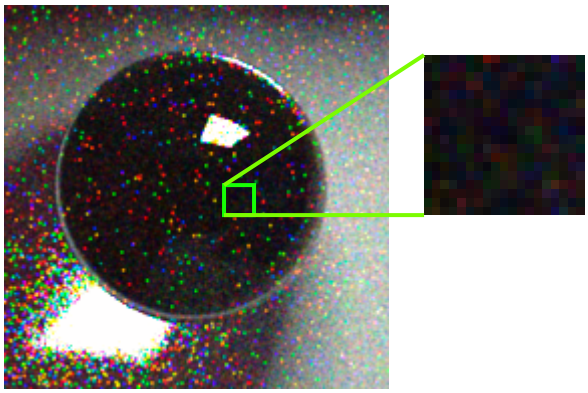

Figure 2: A caustic rendered with way too low sampling density. It is not sufficient to make decisions based on only a few samples, i.e. on small blocks or even pixelwise. The magnified block for example appears to be converged already, but it still misses important features (evident through adjacent bright pixels) and thus needs more samples.

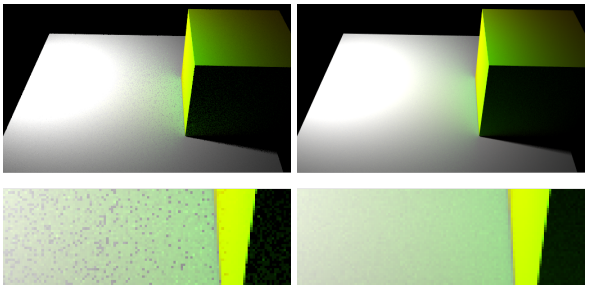

Figure 3: This shows the problem of using a per pixel termination criterion. Due to the nature of Monte Carlo sampling some pixels terminate too early which is clearly visible in the left image. The right image shows our proposed termination criterion with equal sample count.

A block only consists of an axis-aligned bounding box covering an area of the image. The blocks are managed in a linear list. That is, if a block is split, it is simply removed and the two resulting child blocks, disjointly covering the same area of the image, are added to the list again. This way, no explicit hierarchy has to be maintained.

The algorithm continues by drawing new samples (one for each pixel in each remaining block in our implementation). If stratification in image space can be guaranteed, e.g. by using a simple backward path tracer, only non-terminated blocks have to be sampled further. If certain sampling methods cannot be restricted in this way, such as for example light tracing methods, their contribution can still be used, as the stopping condition is not directly coupled to adaptive sampling. In fact, it is also possible to sample adaptively from the lights (see Section 3.2).

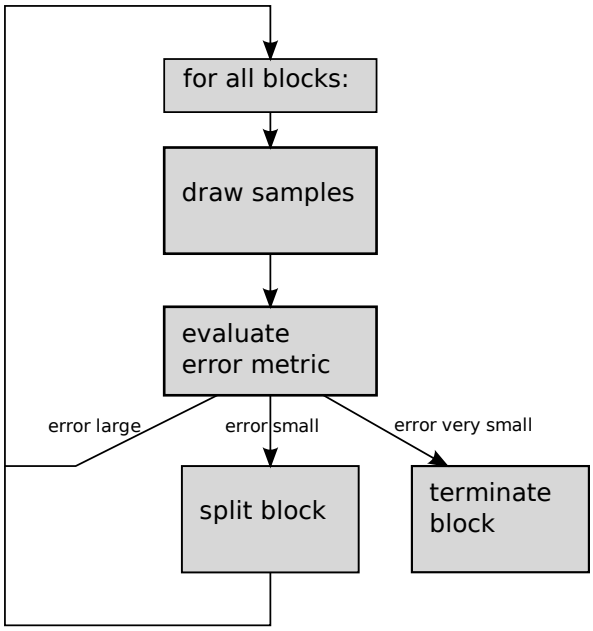

Figure 4: Illustration of a rendering algorithm using the stopping algorithm.

The image is converged when the error of all blocks is smaller than a threshold $\varepsilon_t < \varepsilon_s$. For an overview of the algorithm see Figure 4. An example sequence of blocks is depicted in Figure 1. Note how the algorithm automatically adapts the block size to the image and is thus independent on the resolution.

2.1 Error Metric

We use an error metric based on the pixels in the final image, but to get a robust criterion we always evaluate a block of pixels. We compare two independently computed images with the same number of samples similar to [DW85, DS04].

In the implementation, the evaluation of the error metric can be simplified by introducing just a single additional accumulation buffer and keeping the normal accumulation buffer used to compute the final image. In the second buffer we accumulate only samples every second rendering pass. This is based on the simple observation that computing a single image I with an even sample count can be split into two images A and B with $I = A/2 + B/2 \Rightarrow B/2 - A/2 = I - A$. Thus, using an RGB buffer we estimate the per pixel error as

$$e_p = \frac{|I_p^r - A_p^r| + |I_p^g - A_p^g| + |I_p^b - A_p^b|}{\sqrt{I_p^r + I_p^g + I_p^b}}.$$

The square root in the denominator is motivated by the logarithmic response of the human visual system to luminance. The term here behaves similarly, is easier to evaluate and was found to yield slightly better results.

The error per block is computed by summing over all pixels:

$$e_b = \frac{r}{N} \sum_p e_p$$

where N is the number of pixels contained in this block and r is a scaling factor computed as $\sqrt{A_b/A_i}$. A_i is the area of the image, A_b the area of the block under consideration.

2.2 Block Splitting and Block Termination

The user specifies a single error value v that is used in all further decisions. From this user-specified value we compute two error thresholds ε_s (splitting) and ε_t (termination). For simplicity we use $\varepsilon_t = v$ but it is of course possible to rescale v to a more intuitive parameter range. In all our tests we used $v = 0.0002$. Given ε_t, we compute $\varepsilon_s = 256 \cdot \varepsilon_t$. This is an empirical choice that worked well in all our tested scenes.

The splitting is performed axis-aligned by choosing the axis where the block has the largest extent. The split position is chosen such that the error measure is as equal as possible on both sides.

3 RESULTS

In this section we analyse our proposed stopping criterion for different scenes and additionally examine how well the proposed error metric can be used as an adaptive sampling criterion. In the first subsection we use a path tracer with next event estimation. In Section 3.2 we propose a simple method to extend this adaptive sampling to a light tracing algorithm.

3.1 Image Space

Figure 5 on the left shows a simple test scene of a diffuse sphere illuminated by an area light source casting a large smooth shadow. The right image shows the distribution of the number of samples in the final image as a heat map. White is the maximum number of samples and black the minimum. As each block is sampled uniformly the block structure is still apparent in this image. The graph in Figure 6 shows the number of samples required by the normal rendering algorithm and by our technique to achieve an RMS error below 0.01. In this simple scene the sample count was reduced from $700M$ to $400M$ (57%). This indicates that our error metric is meaningful also with respect to the RMS error. The evaluation of the error metric costs about 10% of the rendering time in our implementation.

As another example we show a living room scene in Figure 7. Again on the left the final image can be seen and on the right the sample count distribution. This is a more realistic scene as it is closed and strong indirect illumination is present. The complexity is distributed almost equally over the whole image. This explains

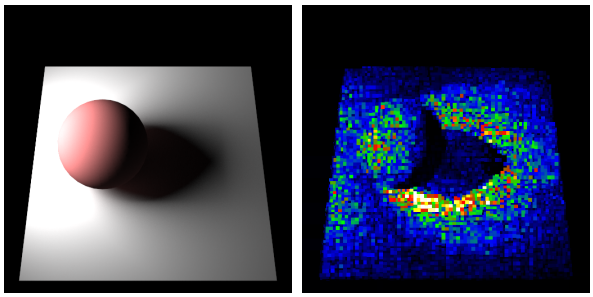

Figure 5: A simple sphere scene rendered with the proposed stopping condition (left) and the respective heat map (right) representing the number of samples required for a converged block. Most effort has to be spent in the penumbra regions.

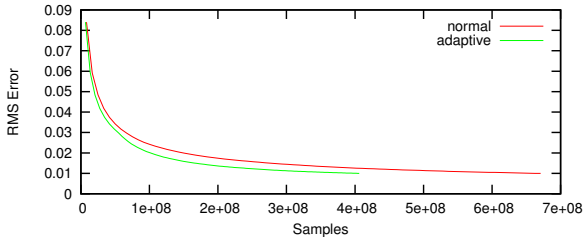

Figure 6: This graph shows the RMS error of the image shown in Figure 5 with increasing number of samples, for normal and adaptive sampling.

why in this scene sampling adaptively does not present a huge benefit. This is also visible in the RMS error graph shown in Figure 8. Still our termination criterion works robustly and also isolates the two small areas in the image where the illumination is more complex due to glossy objects and a small caustic cast by the monkey head on the table (see Figure 9).

3.2 Light Space

In this section we show how to extend our criterion to also be able to adaptively sample in a light tracing setting [Vea97]. A light tracer starts paths only from the light sources and connects hit points to the camera. This is very well suited to compute caustics but it is not possible to directly refine samples on the image plane as it is unknown where exactly a light sample will connect. For our termination criterion this is not a problem as it works equally well independent of the used rendering algorithm. Adaptive sampling of the image plane is not straightforward for light tracing, so all samples need to be distributed equally.

To facilitate adaptive sampling also from the light sources, we back-project the image error metric to an importance map around each light source. Such an importance map consists in a quantized hemisphere around each emitting triangle, thus representing only outgoing directions, ignoring the starting point on the triangle and higher-dimensional bounces. This works well under the assumption that triangles are small with respect to the illuminated area. This can always be achieved by subdividing the emitting triangles which does not change the final image. We chose this approach over explicitly storing the starting point as well to reduce the dimensionality and thus the size of the map. This also simplifies the integration into an existing rendering system. The algorithm proceeds by calculating an initial image calculating a few samples. After that each new light sample accumulates the remaining image error back into its importance map bin at the light source side. This way, the image error metric is back-projected with a delay of one iteration. The importance map can then be used in the next iteration to importance sample light directions (similar to [CAM08]).

As test scene we chose a distorted glass object casting a large caustic (see Figure 10). The top row of Figure 11 shows three importance maps at three different iterations. The bottom row shows the associated image space error blocks. While the top row is heat-map colored by the importance, the colors in the bottom row are solely to distinguish the different blocks. It can be clearly seen how large areas of the hemisphere do not contribute at all to the final image and also how the termination of regions in image space is reflected in the hemispherical importance map. Figure 12 shows the RMS error graph using no adaptivity at all and using the described back-projected importance map. In this case the number of samples required to get an RMS error below 0.01 is reduced from $570M$ light paths to $160M$ light paths (28%).

4 CONCLUSION

We introduced a stopping condition for Monte Carlo image synthesis which automatically adjusts itself to image resolution, robustly adapts to local variance, and is very simple to implement on top of any rendering system. Due to this simplicity it can be easily integrated in GPU based ray tracing systems. The criterion can also be used to facilitate adaptive sampling. Additionally, we have shown how back-projection of the image error metric onto hemispheres around the light sources can be used to profit from adaptive sampling also when starting paths at the light sources.

In the future, it should be possible to extend the system to complete bi-directional light transport. That is, combine adaptive image space sampling and starting the light paths in important directions based on the back-projected image space error at the time and also consider all deterministic connections. It would also be interesting to extend the method to higher dimensionality, similar to Hachisuka et al. [HJW*08], i.e. to remove the dependency on the framebuffer. Furthermore, a mathematical analysis of the variance in path space should be done to gain some knowledge about the worst case integration error at a given sampling density. This way, the initial sampling rate required to assure no fea-

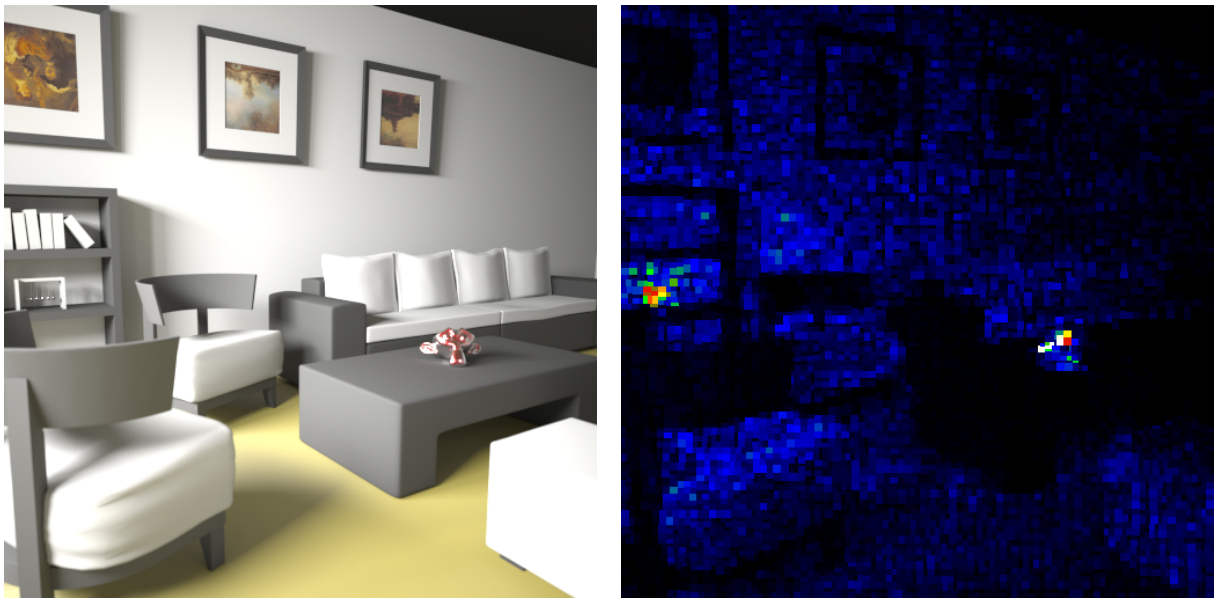

Figure 7: Final image (left) and sample density heat map (right) for the living room scene. The algorithm robustly finds the two spots with significantly higher variance than the rest of the image.

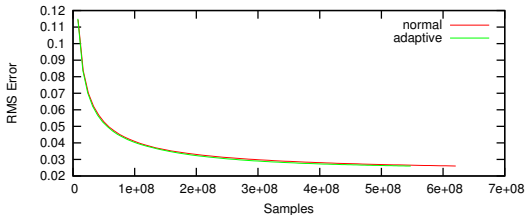

Figure 8: This graph shows the RMS errors of normal and adaptive sampling of the image shown in Figure 7 with increasing number of samples.

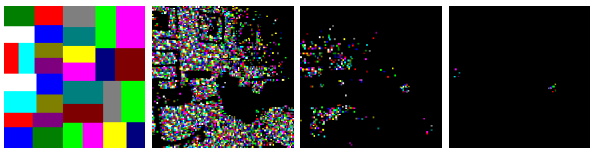

Figure 9: Four images taken from the sequence of active blocks when rendering the living room scene seen in Figure 7. The difficult spots are isolated quickly.

tures are missed when first evaluating the error measure could be determined.

The adaptivity of the method could also be improved upon, if one does not only wish to use it as a termination condition, but also increase efficiency. In analogy to building fast spatial acceleration hierarchies for ray tracing, the image space block splits could be optimized to cut off empty space, i.e. always separate blocks which are likely to terminate in the next iteration.

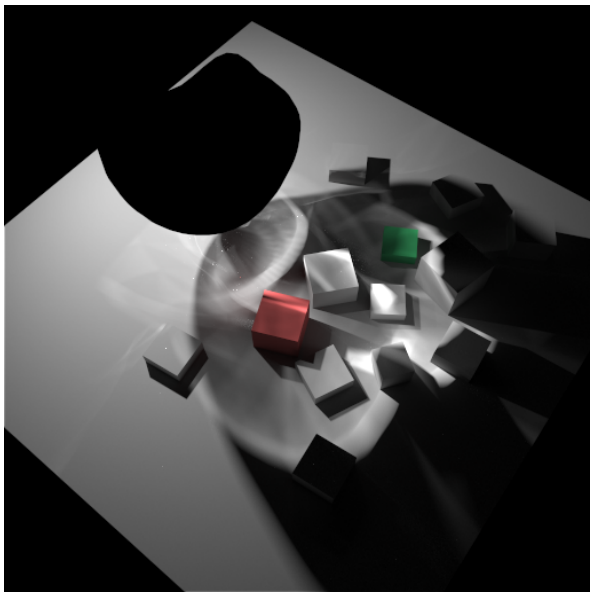

Figure 10: The light tracing test scene. The glass object is black because a pure light tracer cannot deterministically connect singular materials to the camera.

5 ACKNOWLEDGMENTS

This work has been partially funded by mental images GmbH and the DFG Emmy Noether fellowship (Le 1341/1-1).

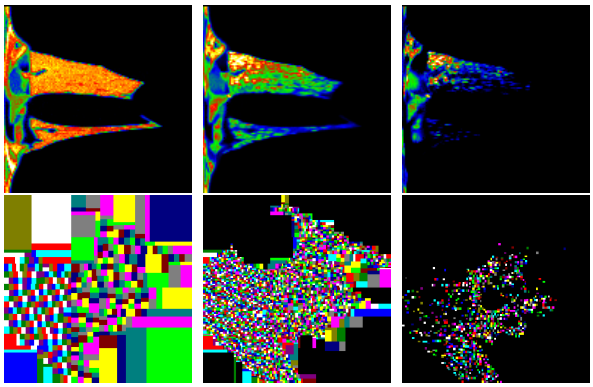

Figure 11: Hemispherical importance map at the light source (top row) and respective image error blocks for the caustic scene (Figure 10).

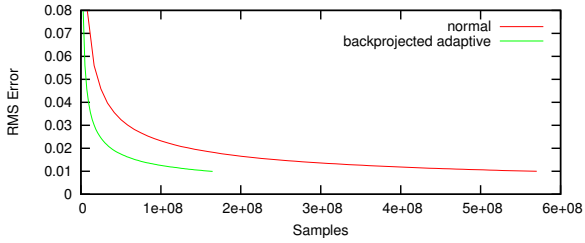

Figure 12: RMS error graph for the light tracing experiment comparing no adaptivity and our proposed back-projection. The curves terminate when the RMS error drops below 0.01.

REFERENCES

[BM98] BOLIN M., MEYER G.: A perceptually based adaptive sampling algorithm. *ACM Transactions on Graphics (Proc. SIGGRAPH 1998)* (1998), 299–309.

[CAM08] CLARBERG P., AKENINE-MÖLLER T.: Practical product importance sampling for direct illumination. In *Computer Graphics Forum (Proc. of Eurographics 2008)* (2008), pp. 681–690.

[Dal93] DALY S.: The visible differences predictor: an algorithm for the assessment of image fidelity. 179–206.

[DS04] DMITRIEV K., SEIDEL H.-P.: Progressive path tracing with lightweight local error estimation. In *Vision, modeling, and visualization 2004 (VMV-04)* (Stanford, USA, 2004), Girod B., Magnor M., Seidel H.-P., (Eds.), Akademische Verlagsgesellschaft Aka, pp. 249–254.

[DW85] DIPPÉ M., WOLD E.: Antialiasing through stochastic sampling. *Computer Graphics (Proc. SIGGRAPH '85)* (1985), 69–78.

[FP04] FARRUGIA J., PÉROCHE B.: A progressive rendering algorithm using an adaptive perceptually based image metric. *Computer Graphics Forum 23* (2004), 605–614.

[HJW*08] HACHISUKA T., JAROSZ W., WEISTROFFER P., DALE K., HUMPHREYS G., ZWICKER M., JENSEN H. W.: Multidimensional adaptive sampling and reconstruction for ray tracing. *ACM Transactions on Graphics (Proceedings of SIGGRAPH '08)* (2008).

[MDMS05] MANTIUK R., DALY S., MYSZKOWSKI K., SEIDEL H.-P.: Predicting visible differences in high dynamic range images - model and its calibration. In *Human Vision and Electronic Imaging X, IS&T/SPIEś 17th Annual Symposium on Electronic Imaging* (2005), Rogowitz B. E., Pappas T. N., Daly S. J., (Eds.), vol. 5666, pp. 204–214.

[Mit87] MITCHELL D.: Generating antialiased images at low sampling densities. *Computer Graphics (Proc. SIGGRAPH '87)* (1987), 65–72.

[Mys98] MYSZKOWSKI K.: The visible differences predictor: applications to global illumination problems. In *Rendering Techniques '98 (Proc. of the Sixth Eurographics Workshop on Rendering)* (1998), pp. 233–236.

[ODR09] OVERBECK R. S., DONNER C., RAMAMOORTHI R.: Adaptive Wavelet Rendering. *ACM Transactions on Graphics (SIGGRAPH Asia 09) 28*, 5 (2009).

[PS89] PAINTER J., SLOAN K.: Antialiased ray tracing by adaptive progressive refinement. *Computer Graphics (Proc. SIGGRAPH '89)* (1989), 281–288.

[PTVF92] PRESS W. H., TEUKOLSKY S. A., VETTERLING W. T., FLANNERY B. P.: *Numerical recipes in C (2nd ed.): the art of scientific computing*. Cambridge University Press, New York, NY, USA, 1992.

[RFS03] RIGAU J., FEIXAS M., SBERT M.: Refinement criteria for global illumination using convex functions. In *Compositional Data Analysis Workshop* (2003).

[RPG99] RAMASUBRAMANIAN M., PATTANAIK S., GREENBERG D.: A perceptually based physical error metric for realistic image synthesis. *ACM Transactions on Graphics (Proc. SIGGRAPH '99)* (1999), 73–82.

[SCCD04] SUNDSTEDT V., CHALMERS A., CATER K., DEBATTISTA K.: Top-down visual attention for efficient rendering of task related scenes. In *In Vision, Modeling and Visualization* (2004), pp. 209–216.

[SGA*07] SUNDSTEDT V., GUTIERREZ D., ANSON O., BANTERLE F., CHALMERS A.: Perceptual rendering of participating media. *ACM Trans. Appl. Percept. 4*, 3 (2007), 15.

[Vea97] VEACH E.: *Robust Monte Carlo Methods for Light Transport Simulation*. PhD thesis, Stanford University, 1997.

[YPG01] YEE H., PATTANAIK S., GREENBERG D.: Spatiotemporal sensitivity and visual attention for efficient rendering of dynamic environments. In *In ACM Transactions on Graphics* (2001), pp. 39–65.