Firefly Detection with Half Buffers

Keith Jeffery
DreamWorks
keith.jeffery@dreamworks.com

ABSTRACT

Fireflies, bright pixels seemingly out of place compared to neighboring pixels, are a common artifact in Monte Carlo ray traced images. They arise from low-probability events, and would be resolved in the limit as more samples are taken. However, these statistical anomalies are often so far out of the expected range that the time for them to converge, even barring numerical instabilities, is prohibitive. Aside from the general problem of fireflies marring a rendered image, their difference in color and variance values can cause problems for denoising solutions. For example, the distance calculation for non-local means filtering [Buades et al. 2005] presented in Rousselle et al. [2012] is not robust under extreme differences in variance.

This paper addresses removing these fireflies to improve both the rendered image on its own, and making the available data more uniform for denoising solutions. This paper assumes a denoising framework that makes use of half buffers and pixel variance, such as set forth in Rousselle et al. [2012]. The variance provides better data than the color channels for determining which pixels do contain fireflies, whereas the half-buffers provide some assurance that the detected firefly is not an expected highlight in the rendered image.

CCS CONCEPTS

• **Computing methodologies** → **Rendering**; **Ray tracing**;

KEYWORDS

ray tracing, mcrt

ACM Reference Format:
Keith Jeffery. 2018. Firefly Detection with Half Buffers. In *Proceedings of ACM SIGGRAPH*. ACM, New York, NY, USA, 2 pages. https://doi.org/10.1145/8888888.7777777

1 PREVIOUS WORK

Mara et al. [2017] address firefly removal in the context of their denoising algorithm, but not as a stand-alone pass. They use a median filter over the image to remove extreme values from the pixel data.

2 INTRODUCTION

Several denoising algorithms make use of half buffers [Bitterli et al. 2016; Rousselle et al. 2012], where the image samples are divided between two buffers, and the variance of the samples for each pixel is saved. The half buffers provide similar images, differing mostly in noise, the mean of which provide the same image that would have resulted in storing all of the samples in a single image. The algorithm set forth in this paper will utilize these two half buffers to detect fireflies.

3 FIREFLY DETECTION

$$\langle I \rangle \approx \frac{1}{N} \sum_{i=1}^{N} \frac{f(\bar{x}_i)}{p(\bar{x}_i)} \tag{1}$$

Equation 1 is the importance-sampling Monte Carlo estimator of the integral $I = \int_\Omega f(\bar{x})\, d\bar{x}$ given N samples and a probability density function, $p(\bar{x})$ from which to draw the samples. As an example of the origin of fireflies, if the ratio between the function value and the probability of sampling that value is large, the estimate of I can be large when not fully converged. Ideally, when the value of $f(\bar{x})$ is large, the value of $p(\bar{x})$ is also large. Issues arise when a low-probability event occurs. These events cause large radiance values, but also increase the variance.

Because of the increased variance, the variance data provide a better source of outlier data than the color channels; some pixels may be much brighter than others in the image, but they may represent an expected and fully-converged highlight.

3.1 Detecting the Fireflies

Fireflies are detected by searching for outliers in the variance data using the *Generalized ESD Test for Outliers* [Rosner 1983]. Unlike other outlier tests, this test does not require knowing the exact number of outliers beforehand, but does require an upper bound, the estimation of which is detailed in Section 3.2.

3.2 Estimating the Upper Bound

Providing an upper bound that is the same as the number of pixels is prohibitively slow, so estimating a tighter upper bound is worthwhile. To estimate the upper bound, the modified Z-score is used [Iglewicz and Hoaglin 1993]. Assuming that the image may not be the final beauty render, black pixels are ignored when estimating the upper bound (the alpha channel can also be used to further rule out pixels). Also, the bound is much tighter if the square root is taken to convert the variance to the standard deviation. This upper bound is then supplied to the Generalized ESD Test for Outliers.

4 FIREFLY REMOVAL AND RECONSTRUCTION

The basic algorithm for reconstruction is straightforward: for every pixel that is an outlier, reconstruct its color value by applying a filter kernel to its neighboring pixels, excluding any neighboring outliers. In testing a box filter, a truncated Gaussian filter, and a Lanczos sinc filter, a Gaussian filter five pixels wide gave the most accurate

ACM SIGGRAPH, August 2018, Vancouver, Canada

ACM ISBN 978-1-4503-1234-5/17/07.
https://doi.org/10.1145/8888888.7777777

Initial	2	2	x	x	8	8
Filtered	2	2	2	2	8	8

Table 1: An example of directional filtering, where x signifies outlier data. A directional filter will bias the result by propagating information in one direction only.

image reconstruction [Smith 1995]. Variance is also updated in this manner for information in a later denoising pass.

Ignoring neighboring outliers can lead to problems with consistency and computability: in the worst case, every neighboring pixel within the kernel footprint is also an outlier, providing no valid data from which to reconstruct the pixel. This happens in wide specular highlights, see the accompanying figure. Making the assumption that the pixels on the edge are valid, the computability problem can be overcome by reconstructing the pixels directionally, e.g. left-to-right: pixels from at least one edge will have already been reconstructed, providing necessary data. However, this can still lead to consistency problems, as the directional pass will propagate values in one direction only. See Table 1.

To address these issues, the reconstruction is done in two phases: the first phase keeps a list of uninitialized outlier pixels and will iterate until they have all been given an initial value. The second phase will iterate over the outlier pixels until a desired amount of convergence is met without having to worry about uninitialized values.

5 HIGHLIGHTS

Even when checking the variance data, legitimate highlights can register as fireflies. For example, the specular samples originating from the same pixel may contribute vastly different luminance values by hitting different portions of an environment light (see Figure 1). Using the half buffers, it can be determined, with high probability, if the bright pixel is a highlight or a random firefly due to a low-probability event. Simply, if the detected outlier exists at the same pixel location in both buffers, it can be assumed that the pixel contains a legitimate highlight, and the bright pixels can be added back to the image. If the firefly removal is a first step to denoising, these pixels can be denoised through non-local means [Buades et al. 2005] before being added back into the output color data.

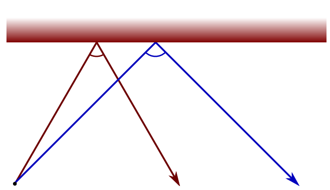

Figure 1: Specular rays from the same pixel can result in vastly different radiance values. The high variance can lead to incorrectly classifying highlights as fireflies.

6 ALGORITHM DETAILS

The most time-consuming part of the algorithm is the Generalized ESD Test for Outliers. Written carefully, with n pixels and with m as the upper bound on the number of outliers, the algorithm can be written in $O(m \log n)$ time. Written in this manner, the most time-consuming part of the Generalized ESD Test is calculating the lambda values, which requires calculating percentage points of Student's t-distribution, which has no closed-form solution. To optimize this portion, results of calculating the percentage points numerically are cached and reused for the second half buffer, and for other frames when temporally filtering. This works because each buffer has the same number of pixels, so the lambda values are invariant over a run.

In the pseudocode accompanying this paper, a minmax heap is used to implement the Generalized ESD Test as it is formulated for two-ended outlier testing. Being interested in only the high variance outliers, this can be simplified to using just a heap. Since no data are being added to the heap after creation, it is also possible to use a sorted array, with a slightly increased complexity of $O(n \log n)$ (since $m \leq n$, and it is likely that $m \ll n$). The same function uses an iterative shifted mean and variance calculation, which allows the computation of the intial statistics in $O(n)$ time, and $O(1)$ addition and removal of a single data point.

7 CONCLUSIONS

Testing half buffers for outliers in the variance data has improved denoising results, and is useful in its own right, even if denoising is not being applied to the images. With careful use of caching and data structure choice, the operation takes less than a second on film resolution frames with no parallelism.

ACKNOWLEDGMENTS

Appreciation is given to DreamWorks and its Research and Development department, especially the MoonRay and MoonShine development teams. Special thanks to Andrew Pearce, Robert Knaack, and Randy Packer for technology leadership.

REFERENCES

Benedikt Bitterli, Fabrice Rousselle, Bochang Moon, José A. Iglesias-Guitián, David Adler, Kenny Mitchell, Wojciech Jarosz, and Jan Novák. 2016. Nonlinearly Weighted First-order Regression for Denoising Monte Carlo Renderings. *Computer Graphics Forum (Proceedings of EGSR)* 35, 4 (June 2016).

Antoni Buades, Bartomeu Coll, and Jean-Michel Morel. 2005. A Non-Local Algorithm for Image Denoising. In *Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Volume 2 (CVPR '05)*. IEEE Computer Society, Washington, DC, USA, 60–65. DOI: http://dx.doi.org/10.1109/CVPR.2005.38

Boris Iglewicz and David Caster Hoaglin. 1993. *How to Detect and Handle Outliers*. ASQC Basic References in Quality Control: Statistical Techniques, Vol. 16. ASQC Quality Press.

Michael Mara, Morgan McGuire, Benedikt Bitterli, and Wojciech Jarosz. 2017. An Efficient Denoising Algorithm for Global Illumination. In *Proceedings of High Performance Graphics*. ACM, New York, NY, USA. DOI: http://dx.doi.org/10.1145/3105762.3105774

Bernard Rosner. 1983. Percentage Points for a Generalized ESD Many-Outlier Procedure. *Technometrics* 25, 2 (May 1983), 165–172.

Fabrice Rousselle, Claude Knaus, and Matthias Zwicker. 2012. Adaptive rendering with non-local means filtering. *ACM Transactions on Graphics (Proceedings of ACM SIGGRAPH Asia 2012)* 31, 6 (Nov. 2012). DOI: http://dx.doi.org/10.1145/2366145.2366214

Alvy Ray Smith. 1995. *A Pixel Is Not A Little Square, A Pixel Is Not A Little Square, A Pixel Is Not A Little Square! (And a Voxel is Not a Little Cube)*. Technical Report. Technical Memo 6, Microsoft Research.