ARPΔ: Hardware-Accelerated Ray-Traced Photon Differentials

Adrian De Barro[a], Keith Bugeja[b] and Sandro Spina[c]
Department of Computer Science, University of Malta, Msida, Malta
{adrian.debarro, keith.bugeja, sandro.spina}@um.edu.mt

Keywords: Photon Mapping, Global Illumination, Ray Tracing Hardware, Photon Differentials, Photon Density Estimation, Photon Footprints, Anisotropic Scattering, Rendering Performance, Ray Differentials, Real-Time Rendering.

Abstract: Photon mapping is a widely used rendering technique that provides biased but consistent global illumination through particle and radius-based density estimation. In this work, we enhance photon mapping by integrating photon differentials, representing each photon as a beam connected to its neighbours. Our proposed method, ARPΔ, dynamically adjusts bandwidth by leveraging changes in both the photon's position and direction, allowing for adaptive control based on the photon's path through the scene. Additionally, ARPΔ combines multiple photon differential strategies to enable efficient global illumination on ray tracing hardware, seamlessly transitioning to progressive photon mapping in highly anisotropic conditions. Experimental results demonstrate that ARPΔ achieves image quality comparable to state-of-the-art photon mapping techniques, validating its effectiveness in producing high-fidelity renders.

1 INTRODUCTION

Photon Mapping and its variants, known for their biased yet consistent solution to the rendering equation, are often preferred over unbiased path-based methods, due to their tendency to produce blurry rather than noisy output until convergence. These methods excel in handling caustics from specular-diffuse-specular ray paths, which are challenging for path-based approaches. While the kernel estimation in photon mapping introduces bias and variance, these can be mitigated with high photon emissions, assuming unlimited memory. Progressive approaches address this by using infinite photon counts within finite memory.

Advancements in GPUs have enabled real-time implementations of photon mapping, leveraging rasterisation pipelines and compute shaders. The work of Evangelou et al. Evangelou et al. (2021) and subsequent extension to progressive photon mapping Kern et al. (2023) have effectively utilised progressive density estimation with ray tracing hardware. By adapting the concept of ray differentials Igehy (1999), photon differentials facilitate adaptive bandwidth selection at the photon level, striking a balance between bias and variance, based on the photon's trajectory.

In this paper, we introduce Hardware-Accelerated Ray-Traced Photon Differentials (ARPΔ), a novel approach to global illumination using photon differentials optimised for ray tracing hardware. The key contributions are:

- the integration of multiple photon differential techniques to achieve global illumination; and
- the adaptation of photon differentials for efficient use on ray tracing hardware.

2 BACKGROUND

Ray differentials Igehy (1999) extend a parametrised ray into a beam by differentiating it, estimating its relationship with neighbouring rays as it propagates. Each ray, r, combines position, p, and direction, d, differentiated with respect to the tangent space described by vectors u and v as:

$$\frac{\partial r}{\partial u} = (\frac{\partial p}{\partial u}, \frac{\partial d}{\partial u}), \quad \frac{\partial r}{\partial v} = (\frac{\partial p}{\partial v}, \frac{\partial d}{\partial v}). \tag{1}$$

Positional and directional derivatives describe tangent space changes and beam angular spread, respectively. Ray differentials are advantageous over Cone and Beam Tracing for maintaining simplicity in tracing

[a] https://orcid.org/0000-0002-3087-9218
[b] https://orcid.org/0000-0002-3111-1251
[c] https://orcid.org/0000-0001-7197-410X

De Barro, A., Bugeja, K. and Spina, S.
ARPÎŤ: Hardware-Accelerated Ray-Traced Photon Differentials.
DOI: 10.5220/0013216800003912
Paper published under CC license (CC BY-NC-ND 4.0)
In *Proceedings of the 20th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISIGRAPP 2025) - Volume 1: GRAPP, HUCAPP and IVAPP*, pages 300-307
ISBN: 978-989-758-728-3; ISSN: 2184-4321
Proceedings Copyright © 2025 by SCITEPRESS – Science and Technology Publications, Lda.

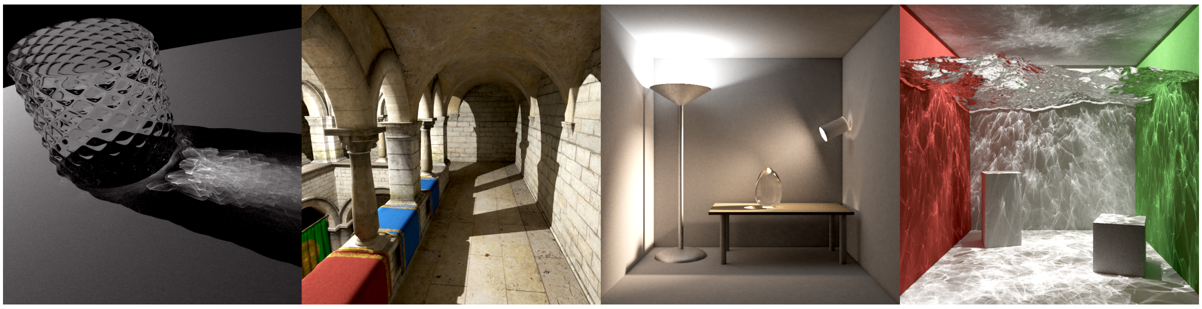

Figure 1: Test scenes rendered using ARPΔ, our locally adaptive hardware ray-traced photon mapper. From left to right: Caustic Glass (C-G), Crytek Sponza (C-S), Veach Bidir (V-B) and Water Caustics (W-C).

procedures as they work seamlessly with standard ray tracing frameworks, relying on simple ray-casting operations without requiring additional geometric complexity or higher-order intersection calculations.

Photon differentials Schjøth et al. (2007) apply ray differentials to density estimation, generating ellipsoidal footprints at intersection points. For point emitters, initial photon area and flux are adjusted upon surface interaction. This formulation only supports point light emitters, with no initial change in positional differentials. On absorption, the photon footprint forms an ellipse, its area derived from the parallelogram cross section formed by the positional differentials. Photon differentials differ from traditional photon mapping by considering all elliptical footprint intersections at a point x_p rather than relying on *k-nearest neighbors* (k-NN) for computing illumination. Intersection tests and irradiance contributions are determined via transformations into filter space, considering the intersected footprints' total radiance at scene points.

3 LITERATURE REVIEW

Photon mapping Jensen (1996) has been widely used for delivering accurate global illumination under different scenarios. For each pixel, it maintains a search radius that limits the maximum distance to search for (using k-NN) contributing photons in the photon map. Photon mapping requires an infinite amount of memory to converge on a solution. Haschisuka et al. Hachisuka et al. (2008) address this limitation by introducing a multi-pass approach for photon mapping, while also progressively reducing the search radius by some factor α. Progressive photon mapping introduces a temporal dependence between frames that was rectified by Knaus et al. Knaus and Zwicker (2011), essentially allowing multiple scattering phases to occur in parallel. Suykens and Willems Suykens and Willems (2001) expanded ray differentials to encompass full ray paths, adding new differentials for sampled events. Initially, Schøjt et al. Schjøth et al. (2007) confined photon differentials to paths that could be represented by ray differentials. This was later broadened by Fabianowski and Dingliana Fabianowski and Dingliana (2009) to incorporate diffuse interreflections and footprint extraction from BVHs. They achieved this without requiring new differentials, instead interpreting diffuse reflections as absorption and re-emission events. Early implementations were limited to point lights, but Frisvad et al. Frisvad et al. (2014) widened the scope of photon differentials to include a variety of light sources. They also introduced screen-space splatting, albeit without addressing diffuse interreflections. Purcell et al. Purcell et al. (2003) were the first to implement an entire photon mapping solution on the GPU. McGuire McGuire and Luebke (2009) employed the GPU for both the initial scattering and final gathering of the photon mapping process. Mara et al. Mara et al. (2013) present various density estimation techniques for screen-space photon mapping.

Smal and Aizenshtein Smal and Aizenshtein (2019) combine the rasterisation pipeline with modern ray tracing hardware to achieve real-time global illumination (GI). Reflective Shadow Maps handle the initial photon scattering, while ray tracing propagates photons through the scene. This method bypasses acceleration structures, splatting indirect illumination directly onto the frame buffer. Kim Kim (2019) employ projection volumes, similar to Jensen's projection map Jensen (2004). These volumes enclose caustic-generating elements, projecting their bounding box opposite to the directional light to form an importance map for caustic generation. Absorbed photons are compressed into single pixels, followed by denoising to diffuse caustic energy. Yang and Ouyang Yang and Ouyang (2021) extend Kim's approach to deliver real-time caustics. They introduce two methods: Adaptive Anisotropic Photon Scattering, which uses photon differentials and adaptive photon emission to generate detailed caustic patterns, and Ray-Guided Water Caustics, tailored for caustics above and below water surfaces. Both approaches rely on the frame buffer for photon contributions.

Evangelou et al. Evangelou et al. (2021) reverse the photon search problem by exploiting acceleration structures provided by hardware ray tracing, and their method is used as the basis for Kern et al.'s Kern et al. (2023) progressive photon mapping implementation. Moreau Moreau and Doggett (2022) leverage Evangelou's inverse-radius approach to accelerate the rendering of indirectly visible caustics. In contrast, techniques like those presented by Yang and Ouyang are unable to render indirect caustics due to their reliance on screen-space methods. Unlike Kern et al., who focus on accelerating photon maps, Moreau and Doggett store pixel-world-space regions of interest, referred to as collection points. During the photon scattering process, photons interacting with diffuse surfaces contribute to the collection points they intersect. This object-space radiance estimation enhances temporal coherence, as the radiance approximation is independent of screen-space constraints.

4 METHOD

Previous work using photon differentials did not provide a comprehensive global illumination solution. Existing solutions either utilised custom acceleration structures or relied on screen-space splatting, foregoing the use of an acceleration structure. Drawing inspiration from the work of Fabianowski et al. Fabianowski (2011); Fabianowski and Dingliana (2009), Frisvad et al. Frisvad et al. (2014), Evangelou et al. Evangelou et al. (2021) and Kern et al. Kern et al. (2023), we leverage the acceleration structures provided by ray tracing hardware to develop ARPΔ, a locally adaptive global illumination technique based on photon differentials.

PPM leverages per-pixel statistics, shrinking the search radius as more photons are detected near a query point in the scene. This mechanism enables PPM to adaptively select bandwidth on a per-pixel basis. Probabilistic Progressive Photon Mapping introduced by Knaus Knaus and Zwicker (2011) moves away from per-pixel information towards a uniform radius for all pixels per iteration, facilitating the parallelisation of the scattering process. Following this, Evangelou et al. and subsequently Kern et al. reverse this process. Instead of employing k-nearest neighbours method to retrieve all photons within a circular region equal to the current global radius, they represent each photon as a sphere with the global radius specified for the respective frame iteration. While this method enhances performance, the approach cannot make use of locally adaptive bandwidth, leading to potentially biased or noisy intermediate results until convergence. For this reason, we have adopted photon differentials. Although our approach does not utilise screen-space statistics, it achieves local bandwidth adaptivity through the resulting change in the differential vectors which represent the photon's footprint at point of intersections.

Footprints that exhibit anisotropy can pose challenges when incorporated into acceleration structures. These may result in excessively large bounding boxes, especially for highly anisotropic footprints that occur when the positional differentials have a small separation angle. To address this issue, problematic footprints are culled and replaced with progressive photons at runtime, ensuring overall performance is maintained.

Table 1 provides definitions for the mathematical notations used throughout this paper, while Figure 2 summarizes the key steps for adopting photon differentials on ray tracing hardware.

Table 1: List of symbols.

Symbol	Definition
α	Angle subtended by two positional differential vectors in tangent space
p	Position vector of a ray or photon
d	Direction vector of a ray or photon
$\frac{\partial}{\partial u}$	Partial derivative with respect to u
$\frac{\partial}{\partial v}$	Partial derivative with respect to v
r	Ray represented as position and direction
x_p	Intersection position vector of a photon on a surface
$\vec{n}_p$	Photon's surface intersection normal
M_p^{-1}	Inverse transform from footprint's tangent space to filter space
M_p	Tangent space transform of the footprint
$\vec{p}$	Position vector of a photon intersection
$\vec{d}'$	New reflection direction vector
v	Virtual travel distance of a photon
t	Travel distance of a photon to the next surface
D_u	Positional differential vector in direction u
D_v	Positional differential vector direction v
ω'	New direction vector in a photon path
$h_{\max}$	Maximum allowed length of the major differential
f	Threshold fraction for degenerate removal
S	Smoothing parameter, a scaling factor applied to photon differential footprints
S_c	Caustics photon map smoothing parameter
S_g	Global photon map smoothing parameter

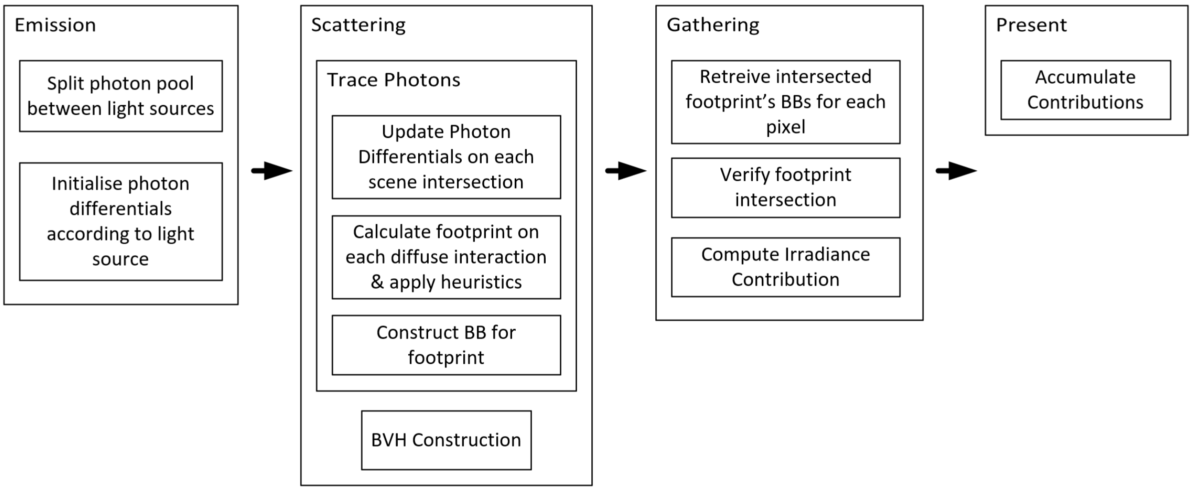
Emission
Split photon pool between light sources
Initialise photon differentials according to light source
Scattering
Trace Photons
Update Photon Differentials on each scene intersection
Calculate footprint on each diffuse interaction & apply heuristics
Construct BB for footprint
BVH Construction
Gathering
Retreive intersected footprint's BBs for each pixel
Verify footprint intersection
Compute Irradiance Contribution
Present
Accumulate Contributions

Figure 2: Pipeline illustrating the primary steps for incorporating photon differentials into ray tracing hardware.

4.1 Emission and Propagation

Our method adopts a framework similar to that of Frisvad et al. Frisvad et al. (2014) for its effective handling of various light sources. However, to achieve a full global illumination solution, we address diffuse interreflections, which are not accounted for in Frisvad et al.'s model. Suykens Suykens (2002) treats each diffuse interreflection as a random event, requiring new differentials for every diffuse reflection. This increases the computational load significantly due to the need for constant updates and management of a list of differentials and the complex calculation of photon footprints as a Minkowski sum.

In order to only track a singular set of photon differentials, we integrate Fabianowski Fabianowski (2011) absorption and re-emission heuristic into Frisvad et al.'s framework. At each photon intersection point $\vec{p}$, we construct a tangent plane for the new reflection direction $\vec{d}'$, setting the length of the tangent and bi-tangent to match those used during the initial light source emission. We calculate the virtual distance v a photon travels before re-emission using:

$$v = \sqrt{\frac{\|D_u x \times D_v x\|}{\|D_\phi \omega' \times D_\theta \omega'\|}}, \qquad (2)$$

where v is a the square root of the ratio of positional differentials at the intersection point to the initial directional differentials on emission. This ensures that upon re-emission, the photon's positional differentials retain the same footprint size as at their previous intersection. The virtual distance v is then added to the travel distance t at the next surface intersection. Interested readers can refer to Fabianowski's work Fabianowski (2011) for further details.

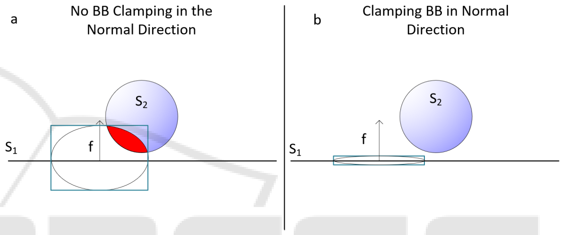
a
No BB Clamping in the Normal Direction
b
Clamping BB in Normal Direction
S_2
S_1
f
S_2
S_1
f

Figure 3: For a photon with an elliptical footprint f, intersecting surface S_1, (a) uses an un-clamped bounding box in the normal direction which results in light leaks to surface S_2 (shaded in red), while (b) clamps the bounding box in the normal direction to reduce the chances of light leaks.

4.2 Photon Map

Two photon maps are used, one for global illumination and the other for caustics. At each diffuse intersection, we add the photon's current state to the acceleration structure. Exploiting the ray tracing hardware's acceleration structures, we treat each footprint as a bounding box (BB). This involves using half the position differentials to determine the BB's extents, thereby identifying its minimum and maximum limits. In practice, these differentials form an ellipsoid, with its maximum height at the footprint's centre being half the minor differential. Since the footprint is ellipsoidal, it might inadvertently contribute to surfaces close by in the normal direction. To prevent this, the maximum BB height along the normal direction is set to the current progressive radius.

4.3 Gathering

Inspired by Evangelou et al. (2021), our gathering process differs from the standard k-NN method by using photon differentials and footprints integrated with

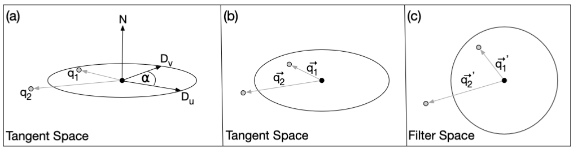

Figure 4: Transformation from tangent space to filter space. (a) In tangent space, the footprint is defined by the normal vector and positional differentials. Vectors connecting the footprint's center to the query points q_1 and q_2 are constnructed. (b) Tangent space is shown along the $-N$ direction for clarity. (c) Vectors $\vec{q_1}$ and $\vec{q_2}$ are transformed into filter space using M_p^{-1}. q_1 intersects the footprint since $\vec{q_1}'$ has a magnitude smaller than one.

the BVH. We construct a ray originating at the scene's query point to detect all intersecting BBs. To confirm the ray's presence within the footprint, we transform the footprint into filter space and then check for an intersection with a unit circle. As suggested by Fabianowski and Dingiliana Fabianowski and Dingliana (2009), this involves computing the inverse matrix of the ellipsoid. Since this is impractical for real-time settings, we adopt Frisvad et al.'s method that directly computes the first two rows of M_p^{-1}:

$$M_p^{-1} = \frac{2}{D_u x_p \cdot (D_v x_p \times \vec{n}_p)} \cdot \begin{bmatrix} D_v x_p \times \bar{n}_p \\ \bar{n}_p \times D_u x_p \end{bmatrix}, \qquad (3)$$

where D_u and D_v are the positional differentials at the intersection point, x_p is the world-space intersection point, $\vec{n}_p$ is the intersection normal and M_p^{-1} is the inverse of matrix M_p, the matrix representing the footprint space. The transform M_p^{-1} maps the anistropic footprint into a unit circle. To determine if a query point intersects a footprint, a vector from the footprint's center to the query point is constructed and transformed into filter space using M_p^{-1}. If the magnitude of this transformed vector is less than or equal to one, the query point resides within the footprint. Figure 4 illustrates the transformation from tangent space to filter space, and demonstrates how intersections with query points are identified.

4.4 Smoothing Parameters

A smoothing parameter is employed to adjust the size of all footprints Schjøth et al. (2007). Initial attempts to use a unified smoothing parameter for both caustic and global photon maps proved ineffective, as caustic and global ellipsoids often demand distinct scaling factors for optimal results. To resolve this, two independent scaling factors S_c (caustics photon map) and S_g (global photon map) are introduced. On absorption, photons are classified as either caustic or global, scaled by the respective smoothing parameter and added to the corresponding photon map.

4.5 Heuristics

Dealing with sizeable ellipsoidal footprints in acceleration structures is difficult. Their limited illumination contribution contrasts sharply with their extensive bounding boxes. This difference often causes ineffective spatial partitioning in the BVH and more frequent point-ellipse intersection tests. Some ellipsoids from caustics generate extended, non-axis aligned footprints, requiring an unduly large BB that substantially increase redundant intersection queries.

To address these shortcomings, we employ four heuristics: *H Max Limit*, *Dampened Adaptation* and *Degenerate Removal* Fabianowski (2011) and *Bounding Box Culling* heuristic adapted from Kaplanyan and Dachsbacher (2013). The *H Max Limit* heuristic clamps the positional differentials of a footprint to a set length h_{max}, while *Dampened Adaptation* raises the major positional differential ∂P to $\partial P^{1/4}$. The minor differential is dampened by the same ratio. *Degenerate Removal* is a heuristic identifying ellipsoids with a footprint height below a small threshold fraction f of their base. Fabianowski Fabianowski (2011) identify and exclude differentials that are degenerate; in ARPΔ they are replaced by a photon whose bandwidth is equal to the current progressive radius. The respective progressive radius is determined according to the current frame index and an initial radius dimension. *Bounding Box Culling* removes any non-degenerate ellipsoids with a bounding box larger than 5% of the total scene area and replaces them with a progressive photon.

5 RESULTS

ARPΔ was developed using Falcor Kallweit et al. (2022), building upon the implementation by Kern et al. Kern et al. (2023). Four different scenes were chosen on the basis of their illumination properties, each with S_c and S_g parameters, obtained experimentally: Caustic-Glass (C-G) Kern et al. (2023) with 0.55 and 1.00, Crytek Sponza (C-S) McGuire (2017) with 1.00 and 0.87, Veach-Bidir Room (V-B) with 0.26 and 0.66 Bitterli (2016), and Water Caustic (W-C) with 0.50 and 0.77. Reference images were generated using RTPM Kern et al. (2023) by accumulating each view for a total of 100,000 iterations and scattering one million photons per iteration. Initial progressive photon mapping radii were set to 0.01 for caustic photons and 0.05 for global photons. These were determined experimentally. Similar to Kern et al. Kern et al. (2023) and Frisvad et al. Frisvad et al. (2014) we evaluate ARPΔ in terms of performance (average iteration

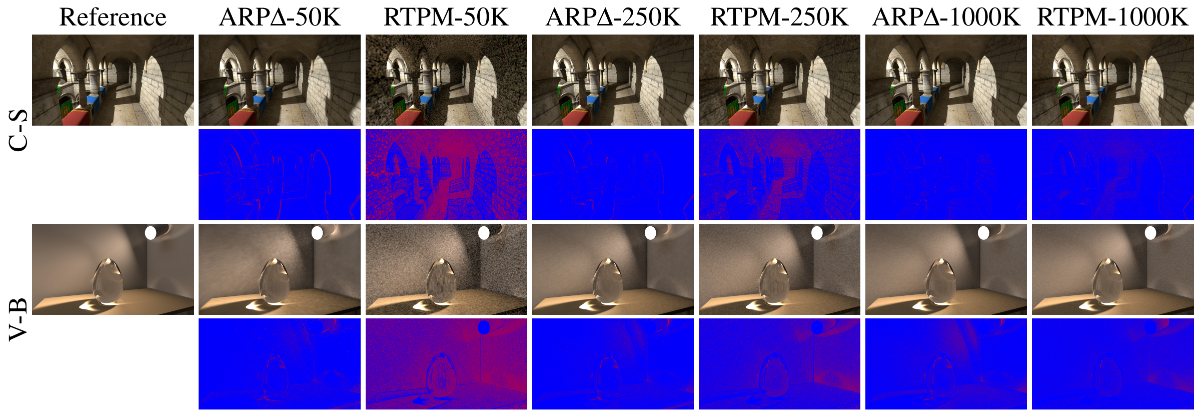

Figure 5: Renders and the corresponding SSIM heatmaps at the ten-second mark for C-S and V-B, for all three experiment configurations (E-50K, E-250K and E-1000K).

Scene	Method	E-1000K			E-250K			E-50K		
		SSIM	NI	ATI (ms)	SSIM	NI	ATI (ms)	SSIM	NI	ATI (ms)
C-G	ARPΔ	0.992	802	12.8	0.993	1485	6.8	0.981	1713	6.5
	RTPM	0.998	1863	5.4	0.997	1832	5.4	0.990	1861	5.5
C-S	ARPΔ	0.988	1449	6.9	0.980	1581	6.7	0.932	1515	6.4
	RTPM	0.986	1668	6.2	0.956	1658	6.1	0.831	1684	5.9
V-B	ARPΔ	0.969	884	11.3	0.974	1535	6.4	0.966	1645	5.8
	RTPM	0.976	1268	7.0	0.938	1790	5.5	0.790	1778	5.6
W-C	ARPΔ	0.989	449	22.7	0.992	971	10.3	0.977	1616	6.1
	RTPM	0.941	1386	6.9	0.993	1819	5.7	0.980	1845	5.3

Table 2: Image quality results under time-normalised constraints (10s) for ARPΔ and RTPM. Iteration counts up to the 10 second mark (NI) and the respective average time per iteration (ATI) in milliseconds, are also reported.

time) and image quality using the SSIM (Structural Similarity Index Measure) metric.

Three experiments, E-1000K, E-250K and E-50K, were conducted to evaluate ARPΔ across a range of photon densities: E-1000K with one million photons per iteration for high-density scenarios, and E-250K and E-50K, with photon counts reduced to a quarter and a twentieth of E-1000K's respectively per iteration, to evaluate performance at lower photon densities. Ten second runs were conducted for the three experimental setups, both with ARPΔ and RTPM (see Table 2). Figure 5 shows the SSIM error heatmaps for the Cytek-Sponza (C-S) and the Veach-Bidir (V-B) scenes. For each scene, the experiments were conducted with uniform smoothing parameters and consistently applied all the heuristics mentioned in §4.5, ensuring standardisation across different scenarios. All results were rendered at a resolution of 1920 × 1080 on an NVIDIA RTX 2080 Ti.

6 DISCUSSION

The results show that our method can achieve accurate global illumination with effective convergence, even at reduced sampling frequencies. We attribute these results to the local adaptive nature of the photon footprints. However, optimising this bandwidth can be computationally demanding, leading to lower frame rates, as observed in the Water Caustics scene during the E-1000K experiment. This scene has dense photon regions and varying AABB sizes which increase the point-ellipsoid queries computed during the gathering process. Variance underneath the glass egg in the Veach-Bidir scene was still noticeable after 10,000 iterations of accumulation (Figure 6, left). This observation corroborates the limitations previously noted by Frisvad et al. Frisvad et al. (2014). Additionally, a minor reduction in SSIM accuracy was noted in the Crytek-Sponza scene, attributable to larger initial differential magnitudes stemming from decreased photon counts. This impacted the precision

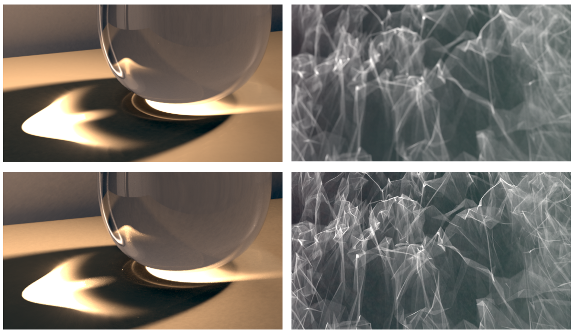

Figure 6: Qualitative comparison of caustics under E-1000K (top-left: RTPM, 100,000 iterations; bottom-left: ARPΔ, 10,000 iterations; right: RTPM and ARPΔ, 500 iterations).

of shadows, as the smoothing parameters remained unchanged.

Photon differentials, in contrast to progressive techniques, enable the rendering of sharp and well-defined caustics, as demonstrated in Figure 6 and Table 2. Progressive methods often result in blurry caustics before reaching convergence, noticeable in the Water-Caustics and Caustic-Glass scenes (see Figure 6, right). While reducing the initial radius can mitigate this blurriness, it also increases variance. By integrating progressive photon mapping with photon differentials, we effectively address the challenge of extremely anisotropic photons by defaulting to progressive photons when such cases are detected. We posit that large photon footprints and their corresponding large bounding volumes hinder balanced BVH construction. Preliminary tests indicate that using the culling heuristic boosts computational efficiency.

7 CONCLUSION & FUTURE WORK

This paper presents a new approach to global illumination using photon differentials optimised for ray tracing hardware. When facing problematic ellipsoid footprints, we switch to progressive photon mapping to ensure the accuracy of our solution. The results demonstrate high-quality physically based rendering at real-time rates. Future research should focus on automating the smoothing parameters based on photon density, illumination type, photon path and scene geometry.

REFERENCES

Bitterli, B. (2016). Rendering resources. https://benedikt-bitterli.me/resources/.

Evangelou, I., Papaioannou, G., Vardis, K., and Vasilakis, A. A. (2021). Fast radius search exploiting ray tracing frameworks. *Journal of Computer Graphics Techniques (JCGT)*, 10(1):25–48.

Fabianowski, B. (2011). *Interactive Manycore Photon Mapping*. PhD thesis, Trinity College (Dublin, Ireland). School of Computer Science & Statistics.

Fabianowski, B. and Dingliana, J. (2009). Interactive global photon mapping. *Computer Graphics Forum*, 28(4):1151–1159.

Frisvad, J. R., Schjøth, L., Erleben, K., and Sporring, J. (2014). Photon differential splatting for rendering caustics. *Computer Graphics Forum*, 33(6):252–263.

Hachisuka, T., Ogaki, S., and Jensen, H. W. (2008). Progressive photon mapping. *ACM Trans. Graph.*, 27(5).

Igehy, H. (1999). Tracing ray differentials. In *Proceedings of the 26th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '99, page 179–186, USA. ACM Press/Addison-Wesley Publishing Co.

Jensen, H. W. (1996). Global illumination using photon maps. In Pueyo, X. and Schröder, P., editors, *Rendering Techniques '96*, pages 21–30, Vienna. Springer Vienna.

Jensen, H. W. (2004). A practical guide to global illumination using ray tracing and photon mapping. In *ACM SIGGRAPH 2004 Course Notes*, SIGGRAPH '04, page 20–es, New York, NY, USA. Association for Computing Machinery.

Kallweit, S., Clarberg, P., Kolb, C., Davidovič, T., Yao, K.-H., Foley, T., He, Y., Wu, L., Chen, L., Akenine-Möller, T., Wyman, C., Crassin, C., and Benty, N. (2022). The Falcor rendering framework. https://github.com/NVIDIAGameWorks/Falcor.

Kaplanyan, A. S. and Dachsbacher, C. (2013). Adaptive progressive photon mapping. *ACM Trans. Graph.*, 32(2).

Kern, R., Brüll, F., and Grosch, T. (2023). Accelerated photon mapping for hardware-based ray tracing. *Journal of Computer Graphics Techniques (JCGT)*, 12(1):1–28.

Kim, H. (2019). *Caustics Using Screen-Space Photon Mapping*, pages 543–555. Apress, Berkeley, CA.

Knaus, C. and Zwicker, M. (2011). Progressive photon mapping: A probabilistic approach. *ACM Trans. Graph.*, 30(3).

Mara, M., Luebke, D., and McGuire, M. (2013). Toward practical real-time : Efficient gpu density estimation. In *Proceedings of the ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games*, I3D '13, page 71–78, New York, NY, USA. Association for Computing Machinery.

McGuire, M. (2017). Computer graphics archive. https://casual-effects.com/data.

McGuire, M. and Luebke, D. (2009). Hardware-accelerated global illumination by image space photon mapping.

In *Proceedings of the Conference on High Performance Graphics 2009*, HPG '09, page 77–89, New York, NY, USA. Association for Computing Machinery.

Moreau, P. and Doggett, M. (2022). Real-time rendering of indirectly visible caustics. In Sousa, A. A., Debattista, K., and Bouatouch, K., editors, *Proceedings of the 17th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications*, volume 1 of *GRAPP*, page 39–48, Setúbal, PRT. INSTICC, SciTePress.

Purcell, T. J., Donner, C., Cammarano, M., Jensen, H. W., and Hanrahan, P. (2003). Photon mapping on programmable graphics hardware. In *Proceedings of the ACM SIGGRAPH/EUROGRAPHICS Conference on Graphics Hardware*, pages 41–50. Eurographics Association.

Schjøth, L., Frisvad, J. R., Erleben, K., and Sporring, J. (2007). Photon differentials. In *Proceedings of the 5th International Conference on Computer Graphics and Interactive Techniques in Australia and Southeast Asia*, GRAPHITE '07, page 179–186, New York, NY, USA. Association for Computing Machinery.

Smal, N. and Aizenshtein, M. (2019). *Real-Time Global Illumination with Photon Mapping*, pages 409–436. Apress, Berkeley, CA.

Suykens, F. (2002). *On Robust Monte Carlo Algorithms for Multi-Pass Global Illumination*. PhD thesis, Department of Computer Science, KU Leuven, Celestijnenlaan 200A, 3001 Heverlee, Belgium.

Suykens, F. and Willems, Y. D. (2001). Path differentials and applications. In *Proceedings of the 12th Eurographics Conference on Rendering*, EGWR'01, page 257–268, Goslar, DEU. Eurographics Association.

Yang, X. and Ouyang, Y. (2021). *Real-Time Ray Traced Caustics*, pages 469–497. Apress, Berkeley, CA.