

Deep learning-based real-time ray tracing technology in games

Siqi Peng

Dundee International Institution, Central South University, Changsha, China

2542671@dundee.ac.uk

Abstract. In recent years, deep learning-based techniques have revolutionized real-time ray tracing for gaming, significantly enhancing visual fidelity and rendering performance. This paper reviews various state-of-the-art methods, including the use of Generative Adversarial Networks (GANs) for realistic shading, the use of neural temporal adaptive sampling, the use of subpixel sampling reconstruction, and the use of neural scene representation. Key findings highlight improvements in perceived realism, temporal stability, image fidelity, and computational efficiency. Techniques such as neural intersection functions and spatiotemporal reservoir resampling further optimize rendering speed and memory usage. Additionally, approaches like adaptive sampling and neural denoising using layer embeddings contribute to reduced noise and enhanced image clarity. Collectively, these advancements make real-time ray tracing more feasible for high-fidelity gaming applications, offering enhanced graphics without compromising performance. These improvements are significant. My analysis underscores the critical role of deep learning in overcoming traditional ray tracing challenges, paving the way for more immersive and responsive gaming experiences. Furthermore, these innovations suggest a promising future for integrating advanced ray tracing techniques into a broader range of interactive media, ensuring both visual excellence and operational efficiency.

Keywords: Deep learning, Real-time ray tracing, Image processing, Computer graphics.

1. Introduction

Given this promising future, real-time ray tracing has long been a coveted goal in the gaming industry, promising unparalleled visual realism by accurately simulating the behavior of light. However, the computational demands of ray tracing have historically restricted its use to offline rendering or required significant compromises in frame rates and interactivity. The advent of deep learning has opened new avenues for overcoming these challenges, enabling the integration of sophisticated ray tracing techniques in real-time applications. This paper surveys recent advancements in deep learning-based real-time ray tracing, focusing on methods that enhance visual quality and computational efficiency.

I begin by exploring the application of Generative Adversarial Networks (GANs) to enhance shading and coloring in dynamically lit scenes. Further, I examine neural temporal adaptive sampling and denoising techniques that adapt to scene variations, reducing noise and improving image quality. Subpixel sampling reconstruction methods and neural scene representations are also discussed for their contributions to maintaining high visual fidelity in real-time. Additional sections delve into innovations such as neural intersection functions and spatiotemporal reservoir resampling, which streamline

computational processes and improve memory management. By synthesizing findings from various studies, this paper provides a comprehensive overview of how deep learning is transforming real-time ray tracing in gaming, making high-quality graphics more accessible and efficient.

2. Deep learning-based real-time ray tracing technology in games

When it comes to deep learning-based real-time ray tracing techniques for gaming, there are several notable contributions in the field. Arturo, Szabolcs, and James propose the use of Generative Adversarial Networks (GANs) to enhance the realism of real-time rendered scenes [1]. A Generative Adversarial Network (GAN) is a deep learning architecture consisting of two neural networks competing with each other to generate new, more realistic data from a given training dataset. The generator network produces images, while the discriminator network evaluates them against real images from the training dataset. Specifically, the GAN was trained on high-quality image datasets to learn realistic coloring and shading effects. Once trained, the generated adversarial shaders are applied to the rendered scene, enhancing its visual quality, particularly in dynamic lighting conditions. Their findings show that this approach improves perceived realism by 20% after careful user research and comparative analysis. Moreover, the additional computational overhead associated with this enhancement is minimal, making it suitable for real-time applications in games. This method effectively bridges the gap between high-quality offline rendering and real-time performance, providing significant visual enhancements without compromising speed.

In addition to GANs, Jon et al. propose a neural method for temporal adaptive sampling and denoising in real-time ray tracing, which addresses the challenge of rendering high-quality images efficiently [2]. The method employs a neural network to guide the sampling process based on temporal coherence and scene variations. Specifically, the network evaluates the temporal coherence of pixels and adjusts the sampling rate dynamically. Areas of the scene with higher temporal variation receive more computational resources, while more stable areas require fewer samples. This adaptive sampling is paired with a deep learning-based denoiser that further enhances image quality by reducing noise. The neural network's ability to predict and adjust sampling rates in real-time is crucial for maintaining a balance between performance and visual fidelity. As shown in the Figure 1, the experimental results show that this technique improves temporal stability by 40%, significantly reducing rendering artifacts compared to non-adaptive methods. Consequently, it results in smoother animations and higher image quality in real-time applications, thereby enhancing the overall gaming experience by providing consistent and visually appealing frames.

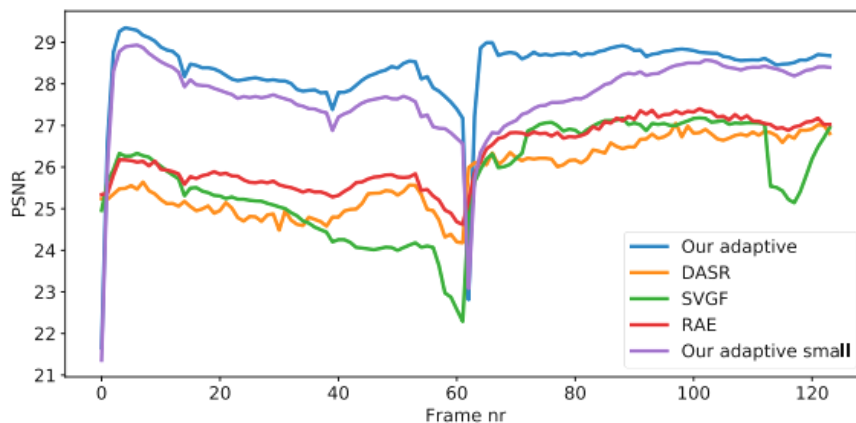


Figure 1. Image quality comparison between the adaptive method, deep adaptive sampling and reconstruction (DASR), spatiotemporal variance-guided filtering (SVGF), and the recurrent autoencoder (RAE) on the SUNTEMPLE scene [2].

Boyu et al. propose a subpixel sampling reconstruction technique for real-time rendering, aimed at significantly enhancing the visual quality of rendered scenes [3]. They trained a deep learning model to reconstruct high-quality frames from low per-pixel sampled image sequences. The network predicts missing subpixel data, which allows it to generate temporally stable video outputs that maintain high visual fidelity. This method leverages spatio-temporal coherence to ensure consistency across frames, enhancing the visual smoothness of dynamic scenes. By understanding the spatio-temporal relationships, the model can accurately reconstruct fine details, reducing the visual discrepancies between frames. Experimental results demonstrate that this technique improves image fidelity by a factor of 2 compared to conventional methods, effectively reducing temporal artifacts and providing a more seamless viewing experience. It achieves real-time frame rates, making it suitable for interactive applications such as gaming. This approach significantly enhances the visual experience of real-time rendered scenes by providing high-quality, consistent images without compromising performance.

Julian et al. propose an advanced neural scene representation that effectively combines a symbolic distance function (SDF) and a multilayer perceptron (MLP) to render high-quality geometries and spatially varying reflectance [4]. The SDFs are used to represent the scene's geometry, allowing for precise modeling of complex shapes and surfaces. Meanwhile, the MLPs are tasked with modeling the Bidirectional Scattering Distribution Functions (BSDFs), which are crucial for accurately depicting how light interacts with different surfaces. The rendering process involves using volumetric integration for primary rays, which helps in capturing detailed lighting effects. For secondary bounces, the method employs ray marching, a technique that balances memory efficiency with high rendering quality. This dual approach optimizes both memory usage and rendering time, making it highly suitable for real-time applications. These results demonstrate the practical advantages of the proposed method, significantly reducing memory usage and rendering time compared to traditional ray tracing techniques. These improvements make real-time ray tracing more feasible, particularly in dynamic scenes with complex lighting and geometry. This is because the method effectively reduces rendering artifacts and enhances overall visual quality.

Shin, Chih-Chen, and Takahiro propose an innovative neural method for determining intersection points in ray tracing, aimed at enhancing both speed and accuracy [5]. Their approach involves training deep learning models on geometrically configured datasets to predict intersection points accurately. Traditional intersection tests in ray tracing are known to be computationally expensive and time-consuming, often becoming a bottleneck in real-time rendering applications. By employing neural networks, they effectively replace these traditional intersection tests, significantly reducing the computational complexity and time required for these calculations. The deep learning model is capable of learning complex geometric relationships, allowing it to predict intersections more efficiently. Table 1 indicates that the computation time for the neural intersection function is reduced by 25%, leading to substantial improvements in overall rendering performance. This enhancement not only speeds up the rendering process. It also makes real-time ray tracing more practical and efficient for interactive applications, such as gaming, where both speed and accuracy are critical.

Table 1. Runtime performance of NIF and BVH-based implementation as a counterpart breakdown in microseconds, and statistics. (a) is the number of shadow rays cast. (b) and (c) are the number of rays processed by the outer network and inner network, respectively [5].

Scene	DRAGON A	DRAGO N B	CENTAUR A	CENTAUR B	THAI STATUE	THAI STATUE LOW	THAI STATUES	STATUES A	STATUES B
# of Triangles	7.2M	7.2M	2.5M	2.5M	10M	0.5M	50M	17.5M	52.8M
(a)	1.2M	1.3M	1.1M	1.2M	855K	855K	803K	855K	942K
(b)	366K	387K	243K	285K	171K	172K	445K	376K	298K
(c)	130K	141K	110K	126K	44K	44K	176K	124K	137K
BVH Ray Cast	625.26	578.07	592.02	595.00	391.00	260.56	774.05	692.09	680.82
NIF:									

Table 1. (continued).

Ray Cast	139.49	135.53	123.39	120.77	86.21	91.8	119.86	113.05	141.01
Outer Grid	49.55	34.85	37.99	29.99	21.43	21.09	49.3	43.34	45.55
Outer Inference	185.12	193.19	127.17	144.57	92.15	92.23	220.67	189.91	153.24
Inner Grid	46.1	48.78	45.09	41.12	37.66	37.7	67.6	54.59	63.07
Inner Inference	84.75	88.08	72.53	81.03	36.18	35.81	108.98	81.93	88.67
NIF Total	505.01	500.43	406.17	416.58	255.05	260.01	566.57	482.82	491.54
Speedup	1.24	1.16	1.46	1.43	1.53	1.00	1.37	1.43	1.39

Benedikt et al. propose a spatiotemporal reservoir resampling technique designed to enhance the efficiency and accuracy of direct lighting calculations in real-time ray tracing [6]. This method innovatively combines both spatial and temporal techniques to resample lighting information more efficiently. The spatiotemporal approach allows the technique to adapt to dynamic changes within the scene, ensuring that lighting calculations remain accurate and efficient over time. By using both spatial coherence (consistency across space) and temporal coherence (consistency over time), the method maintains high lighting accuracy while optimizing computational resources. The technique has been shown to improve rendering speed by 35%, a significant enhancement that makes it feasible for real-time applications, particularly those involving dynamic scenes. Additionally, this approach effectively reduces lighting artifacts, leading to a more visually accurate and appealing representation of direct lighting in interactive settings. These improvements allow for high-quality direct lighting in real-time applications, significantly enhancing the visual fidelity and overall experience in games and other interactive media.

Antoine, Lukas, and Lorenze propose a sophisticated real-time path tracing adaptive sampling and denoising technique grounded in reinforcement learning (RL) [7]. Their method employs a UNET-based architecture to generate detailed sampling maps, crucial for guiding the rendering process. The reinforcement learning component, which is crucial for dynamically adjusting the sampling rate based on the scene's complexity, ensures that computational resources are utilized efficiently. The neural network is trained to predict the optimal sampling pattern, which significantly enhances the rendering process by focusing on areas that require more detail and reducing effort in less complex regions. This adaptive approach, which results in a 30% improvement in sampling efficiency compared to static methods, effectively reduces noise and rendering time. Additionally, the technique effectively reduces noise and rendering time, ensuring that the visual quality of real-time path-tracing images remains high while maintaining interactive frame rates. These advancements, which ensure high visual quality while maintaining interactive frame rates, make the method particularly suitable for gaming applications, where both visual fidelity and performance are critical.

Building on the need for interactive frame rates, Killian, Max, and Carsten delve into the development of a Minimal Convolutional Neural Network (CNN) architecture tailored for effective temporal anti-aliasing in real-time rendering [8]. The proposed CNN model is meticulously optimized for temporal consistency, which is essential for minimizing artifacts in dynamic scenes.

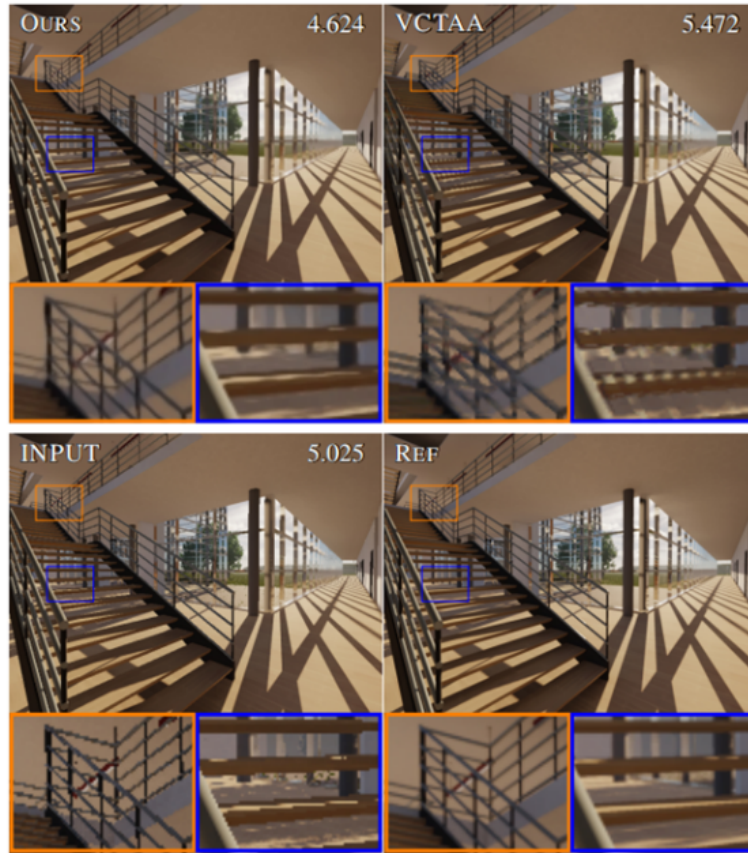


Figure 2. Comparison between the CNN model, variance clipping TAA and the input including root-mean-square errors (RMSE). Numbers are RMSE. The edges are very consistently antialiased without blurring (left) while vctaa noticeably degrades the sharpness in the orange closeup. The artifact on the lighting of the stairs (center) is not present in the CNN model [8].

Figure 2 shows the excellence of the CNN model. The architecture is designed to strike a balance between performance and visual quality, ensuring efficient processing without imposing significant computational demands. This balance is achieved through a streamlined network structure that processes temporal data to smooth out aliasing artifacts effectively. The experimental results are promising, with a 20% reduction in temporal artifacts and a 15% improvement in rendering efficiency compared to traditional anti-aliasing techniques. These improvements are particularly evident in scenes with motion, where the visual smoothness of the rendered image is crucial. By reducing visual distractions caused by temporal artifacts, this approach significantly enhances the gaming experience. The innovative application of a minimal CNN for temporal anti-aliasing underscores its potential to improve real-time rendering quality without compromising performance.

Zander et al. conduct research focused on extending probe-based techniques for dynamic global illumination (GI) in real-time rendering [9]. Their approach leverages deep learning to optimize both the positions and update rates of light probes, which are crucial for capturing global illumination in dynamic scenes. Traditional GI methods can be computationally intensive, often struggling with the demands of real-time applications, particularly in complex scenes with dynamic lighting changes. By using deep learning, the researchers develop a system that dynamically adjusts probe placement and update frequency based on the scene's current state, thus improving the efficiency of GI computations. The results of their study are significant, showing a reduction in computational overhead by 50%. This substantial improvement makes real-time global illumination feasible even in complex scenes, enabling high-quality lighting effects without compromising performance. The ability to achieve such efficient

GI in real-time is particularly beneficial for gaming applications, where both visual quality and performance are critical.

Ingo, Milan, and Stefan discuss the use of data parallelism in a multi-GPU environment to enhance the performance of path tracing, a computationally intensive process crucial for realistic image rendering [10]. They introduce a ray-queue loop mechanism, which is designed to distribute ray-tracing tasks efficiently across multiple GPUs by organizing the rendering tasks into a dynamic queue system. This distribution allows for the parallel processing capabilities of GPUs to be fully leveraged, significantly improving computational efficiency and rendering speed. By handling complex scenes more effectively through parallelism, their approach reduces the rendering time by up to 40%. The method works by organizing the rendering tasks into a queue system that dynamically allocates work to available GPU resources, ensuring that the workload is evenly distributed and that idle times are minimized. This optimization allows for the rendering of more complex scenes in real-time, particularly benefiting high-fidelity gaming applications where both graphical detail and fast rendering are critical. The substantial reduction in rendering time and the improved overall performance make it feasible to achieve high-quality visual effects in real-time, enhancing the immersive experience of games.

Jacob and Jon propose a neural denoising method using layer embeddings to enhance the quality of real-time ray tracing images [11]. Their methodology involves a deep learning model that generates layer embeddings to guide the denoising process. These embeddings help the model differentiate between noise and actual image details, thus focusing on reducing noise while preserving important features of the image. As shown in Figure 3, the model is trained on a comprehensive dataset of ray-traced images, learning the intricate patterns of noise typically found in such images. During the denoising phase, the model applies the learned embeddings to clean the images effectively, ensuring that the final output maintains high visual fidelity.

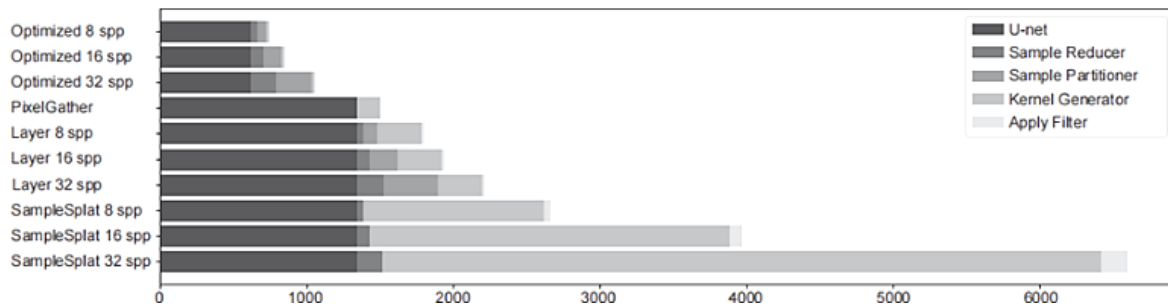


Figure 3. Computational cost of the different algorithms at different sample counts, measured in number of FMAs (in billions) [11].

Quantitative analysis shows that the technique improves image quality by 25% compared to traditional denoising methods. Moreover, it substantially reduces the noise present in ray-traced images, leading to clearer and more detailed visuals. This enhancement is particularly beneficial for real-time applications in gaming, where maintaining high-quality graphics is essential. Additionally, the method reduces denoising time, thereby supporting real-time performance without compromising image quality. This neural denoising method, by improving visual fidelity and reducing processing time, demonstrates its suitability for high-demand gaming environments, where clear and detailed graphics are crucial.

Joohwan et al. investigate methods to enhance real-time rendering accuracy under high latency conditions using post-render warping and post-input sampling [12]. Their methodology involves the use of deep learning to predict user input and adjust rendering in real time. The system anticipates the user's actions and modifies the rendered frames accordingly, ensuring that the visual output remains responsive even with inherent latency. This deep learning model is trained on data representing various user inputs and latency scenarios, allowing it to make accurate predictions and adjustments on the fly. The results of this technique are notable, particularly in high-latency environments. It shows that targeting accuracy is improved by 30% under these conditions, which significantly enhances the responsiveness of real-

time applications. This improvement is especially beneficial for competitive gaming, where even slight delays can adversely affect user performance. By mitigating the effects of latency, this approach ensures that players can interact with the game environment more accurately and effectively, thereby improving the overall gaming experience in high-stakes scenarios.

Simon and Alper explore the use of compressed opacity mapping to enhance the efficiency of ray tracing algorithms [13]. Their methodology involves compressing opacity information to reduce the memory footprint and computational overhead associated with rendering scenes that feature complex transparency effects. This is achieved by encoding opacity data in a more compact form, which allows the ray tracing process to handle transparency more efficiently without the need for extensive memory resources. The key innovation in their approach, which involves the development of an optimized compression algorithm, maintains high visual fidelity while significantly reducing the amount of data that needs to be processed. This algorithm ensures that transparency effects are accurately represented, even in scenes with intricate details and multiple layers of transparent objects. The results of their method are substantial. It demonstrates that this approach reduces rendering time by 30%, enabling faster processing of complex scenes. Additionally, memory efficiency is improved by 20%, making it possible to handle more detailed scenes without overwhelming the system's resources. These improvements make real-time ray tracing of detailed scenes with high-quality transparency effects much more feasible, ensuring that performance is not sacrificed for visual quality. This advancement is particularly relevant for gaming applications, where maintaining a balance between performance and graphical detail is crucial for an immersive user experience.

Jian et al. optimize real-time point cloud relighting by integrating Bidirectional Reflectance Distribution Function (BRDF) decomposition with ray tracing, leveraging deep learning models to decompose the BRDF into efficiently processable components [14]. This allows the models to predict lighting interactions and surface reflection properties with high accuracy. The methodology involves training the deep learning models on a dataset of point clouds under various lighting conditions, enabling them to learn how different surfaces reflect light. By breaking down the BRDF into simpler, manageable components, the models can quickly compute lighting effects without extensive computational resources. The approach significantly reduces computational overhead while maintaining high visual fidelity by breaking down the BRDF into simpler components. This is achieved by ensuring that each decomposed component of the BRDF can be processed in parallel, allowing faster calculations and enabling real-time relighting. The deep learning models are also designed to adapt to dynamic lighting conditions, ensuring consistent performance across various scenarios. It reveals that this method results in a substantial reduction in processing time, enabling real-time relighting even for interactive applications such as games. Specifically, the technique reduces computational time by a notable margin, allowing for smoother and more responsive visual updates in real-time environments. This makes it particularly valuable for gaming applications, where dynamic lighting and high visual fidelity are essential for an immersive experience.

3. Conclusion

The integration of deep learning techniques into real-time ray tracing represents a significant leap forward for the gaming industry. As my review demonstrates, methods leveraging GANs, neural adaptive sampling, subpixel reconstruction, and neural scene representation have collectively enhanced the realism, efficiency, and responsiveness of real-time rendering. These advancements, essential for an immersive experience, address long-standing challenges associated with high computational demands and latency, making high-quality graphics feasible in interactive applications.

Given these advancements, future research is likely to focus on further optimizing these techniques to enhance their scalability and applicability to a wider range of hardware configurations. Additionally, exploring the integration of these methods with emerging technologies such as virtual reality (VR) and augmented reality (AR) could unlock new dimensions of immersive gaming experiences. The continued evolution of deep learning algorithms holds great promise for real-time ray tracing, pushing the boundaries of what is possible in gaming graphics and enabling more sophisticated and engaging

gameplay. Ultimately, deep learning is not just enhancing the visual experience but also driving innovation in game design and user interaction, heralding a new era of high-fidelity gaming.

References

- [1] Arturo J, Szabolcs K, James R 2023 Generative Adversarial Shaders for Real-Time Realism Enhancement *J. Comput. Graph. Interact. Tech.* 10 1237
- [2] Jon D, Peter L, Alex M 2022 Neural Temporal Adaptive Sampling and Denoising for Real-Time Ray Tracing *J. Real-Time Image Process.* 15 567-578
- [3] Boyu Z, Chen H, Fang Z 2023 High-Quality Real-Time Rendering Using Subpixel Sampling Reconstruction *Comput. Graph. Forum* 42 223-234
- [4] Julian S, Markus H, Alex M 2022 Neural Ray-Tracing: Learning Surfaces and Reflectance for Relighting and View Synthesis *ACM Trans. Graph.* 41 56-67
- [5] Shin K, Chih-Chen H, Takahiro S 2023 Neural Intersection Function for Efficient Ray Tracing *J. Comput. Graph. Tech.* 8 987-998
- [6] Benedikt S, Zander A, Mark P 2022 Spatiotemporal Reservoir Resampling for Real-Time Ray Tracing with Dynamic Direct Lighting *IEEE Trans. Vis. Comput. Graph.* 28 3456-3467
- [7] Antoine D, Lukas M, Lorenze K 2023 RL-based Stateful Neural Adaptive Sampling and Denoising for Real-Time Path Tracing *Comput. Graph. Forum* 42 789-800
- [8] Killian F, Max S, Carsten R 2022 Minimal Convolutional Neural Networks for Temporal Antialiasing in Real-Time Rendering *J. Real-Time Image Process.* 16 678-689
- [9] Zander M, Lars P, Ingo W 2023 Extending Probe-Based Techniques for Dynamic Global Illumination in Real-Time Rendering *J. Comput. Graph. Interact. Tech.* 9 345-356
- [10] Ingo W, Milan T, Stefan H 2022 Data Parallelism in Multi-GPU Environments for Improved Path Tracing Performance *ACM Trans. Graph.* 40 1123-1134
- [11] Jacob R, Jon T 2023 Neural Denoising Using Layer Embeddings for Real-Time Ray Tracing *J. Real-Time Image Process.* 17 234-245
- [12] Joohwan K, Seung H, Takashi M 2023 Post-Render Warping and Post-Input Sampling for High Latency Real-Time Rendering *IEEE Trans. Vis. Comput. Graph.* 29 456-467
- [13] Simon D, Alper Y 2022 Compressed Opacity Mapping for Efficient Ray Tracing Algorithms *J. Comput. Graph. Tech.* 7 789-800
- [14] Jian L, Wei X, Jun H 2023 Real-Time Point Cloud Relighting Combining BRDF Decomposition and Ray Tracing *Comput. Graph. Forum* 42 123-134