مجلــــة كليـــة التربيـــة طرابلـــــس

Journal of the Faculty of Education Tripoli

إعادة بناء أجزاء الصور (RGB) بأسلوب هجين مبني على المنطق الكمي باستعمال Mitsuba 3 وQiskit

[1] Ahmed All Elhoni,[2] Ramzi Hamid Elghanuni.

[1,2] Department of Internet Technologies, Faculty of Information Technology, University of Tripoli, Tripoli, Libya.

*Corresponding: a.elhoni@uot.edu.ly

Article history

Received: 28 Jun 2026 | Accepted: 06 Jul 2026 | Published: 26 Aug 2026

الملخص:

يعمل تتبع المسار (Path tracing) على إنتاج صور واقعية فيزيائياً من خلال تقدير انتقال الضوء عبر أخذ عينات "مونت كارلو". لكن الحصول على جودة عالية يتطلب عدداً كبيراً من العينات لكل بكسل. وتقدم هذه الورقة معياراً (Benchmark) هجيناً متواضعاً يربط بين التصيير الفيزيائي (physically based rendering) في نظام " Mitsuba 3" وتجارب الدوائر الكمومية في "Qiskit". تم تصيير مشهد "Cornell Box" كلاسيكياً، واستخراج منطقة اهتمام (ROI) بحجم 8 × 8 بكسل من صورة مرجعية ذات 256 عينة لكل بكسل، حيث تم ترميز قيمة كل قناة لونية (الأحمر، الأخضر، والأزرق) باعتبارها احتمالية قياس في كيو بت (qubit) واحد باستخدام دوران "Ry". تم تقييم مقدِّرين (estimators): أخذ عينات "بيرنولي" المباشر للكيو بت المُجهز، ودائرة تضخيم على نمط "غروفر" (Grover-style) متبوعة بعكس الاحتمالية. وأظهر المقدِّر البسيط (Naive) سلوكاً متوقعاً، حيث انخفض متوسط مربع الخطأ (MSE) للألوان RGB من 0.00480 عند 32 محاولة (shots) إلى 0.000124 عند 1024 محاولة. في المقابل، ظل المقدِّر المُضخَّم الحالي بالقرب من 0.036 كمتوسط مربع خطأ (MSE) عبر كافة عدد المحاولات، مما يشير إلى أن دائرة التضخيم المختبرة أو عملية عكس الاحتمالية غير مناسبة لمهمة إعادة بناء القنوات القياسية الحالية. وبناءً عليه، يساهم هذا العمل في تقديم معيار قابل للتكرار يربط بين "Mitsuba" و"Qiskit" ويوفر نتيجة تشخيصية سلبية، بدلاً من تقديم أدلة على وجود تتبع أشعة كمومي كامل أو تسريع عملي في التصيير.

الكلمات المفتاحية: Mitsuba 3، Qiskit، التصيير المستوحى من الكم، تتبع المسار، إعادة بناء RGB، تقدير السعة، تضخيم "Grover"، صندوق كورنيل (Cornell Box).

Hybrid Quantum-Inspired RGB Patch Reconstruction Using Mitsuba 3 and Qiskit

A B S T R A C T:

Path tracing produces physically plausible images by estimating light transport through Monte Carlo sampling, but high-quality renders require many samples per pixel. This paper presents a modest hybrid quantum-classical benchmark that connects physically based rendering in Mitsuba 3 with quantum-circuit experiments in Qiskit. A Cornell Box scene is rendered classically, an 8 x 8 RGB region of interest is extracted from a 256 samples-per-pixel reference image, and each red, green, and blue channel value is encoded as a one-qubit measurement probability using an Ry rotation. Two estimators are evaluated: direct Bernoulli sampling of the prepared qubit and a Grover-style amplified circuit followed by probability inversion. The naive estimator behaves as expected, with RGB MSE decreasing from 0.00480 at 32 shots to 0.000124 at 1024 shots. In contrast, the current amplified estimator remains near 0.036 RGB MSE across shot counts, indicating that the tested amplification circuit or inversion is not suitable for the present scalar-channel reconstruction task. The work therefore contributes to a reproducible Mitsuba-Qiskit benchmark and a diagnostic negative result, rather than evidence of full quantum ray tracing or practical rendering speedup.

Keywords: Mitsuba 3, Qiskit, quantum-inspired rendering, path tracing, RGB patch reconstruction, amplitude estimation, Grover-style amplification, Cornell Box.

1. Introduction.

Physically based rendering simulates the transport of light through virtual scenes to generate realistic images. Path tracing is widely used for this purpose, but it relies on repeated stochastic sampling of light paths. Increasing the number of samples reduces noise, yet it also increases computational cost (Huo & Yoon, 2021; Yan et al., 2025).

Quantum computing has motivated research into rendering-related subproblems because quantum search and amplitude-amplification methods can reduce the query complexity of selected search or sampling tasks (Martyn et al., 2021; Plekhanov et al., 2021). Work on quantum ray tracing has considered visibility and primitive-search acceleration, while quantum ray marching has explored broader light-transport formulations (Mosier et al., 2023; Santos et al., 2022). These directions are important, but they remain far from routine use in practical rendering pipelines (Mosier et al., 2023; Santos et al., 2022).

This paper takes a narrower and more reproducible route. Mitsuba 3 performs the actual rendering, and Qiskit is used only to study quantum-circuit estimators for compact image-derived scalar values. The experiment asks whether a small rendered RGB patch can be reconstructed from independently estimated amplitude-encoded channel values, and whether a simple Grover-style amplified estimator is beneficial under this setup (Plekhanov et al., 2021).

The answer is intentionally limited. The naive estimator reconstructs the patch increasingly well as shots increase. The tested amplified estimator does not improve reconstruction and instead shows a persistent bias/error pattern. This result is useful because it identifies a concrete failure mode that must be resolved before stronger claims about amplification in this workflow can be made (Plekhanov et al., 2021; Santos et al., 2022).

1.1 Contributions

- A reproducible Python/Jupyter workflow connecting Mitsuba 3 rendering outputs with Qiskit circuit experiments.
- An RGB ROI protocol in which rendered red, green, and blue channel values are encoded as one-qubit measurement probabilities.
- A direct comparison between naive quantum sampling and a Grover-style amplified estimator for RGB patch reconstruction.
- A diagnostic result showing that the current amplified estimator underperforms naive sampling and requires correction or redesign before it can support an improvement claim.
- A complete evaluation structure using RGB patch figures, circuit figures, MSE/MAE tables, per-channel metrics, and error maps.

2. Related Work

Mitsuba 3 is a research-oriented rendering system for forward and inverse light transport simulation. Its Python interface supports reproducible scene construction and rendering through dictionary-based scene loading and functions such as mi.render.

Quantum ray tracing research has studied how quantum search can accelerate selected ray-primitive visibility queries (Santos et al., 2022; Lu & Lin, 2022a). Towards Quantum Ray Tracing reports a quadratic query-complexity improvement for unordered primitive search, but this does not immediately translate to speedup over modern acceleration structures in practical renderers (Santos et al., 2022; Mosier et al., 2023).

Quantum Ray Marching reformulates light transport using quantum random walks (Mosier et al., 2023). This line of work indicates that rendering can be studied in a quantum computational setting, though such algorithms are not yet drop-in replacements for classical path tracers (Mosier et al., 2023; Santos et al., 2022).

Alves, Santos, and Bashford-Rogers (2019) presented one of the earliest rendering-specific quantum algorithms by applying Grover-style search to ray casting from an orthographic camera. Their work focuses on visibility and occlusion rather than full global illumination, but it is important because it frames ray casting as a searchable rendering subproblem and shows how quantum search can be connected to a classical graphics operation.

Lu and Lin (2022a) proposed a broader framework for quantum ray tracing in which sampling and color estimation are expressed using quantum registers and quantum counting. This work is more theoretical than the present benchmark, but it is relevant because it treats ray tracing as a high-dimensional estimation problem and motivates the use of quantum estimation ideas for rendering-related quantities.

Lu and Lin (2023) further studied improved quantum supersampling for quantum ray tracing, replacing the original phase-estimation component with a more robust quantum counting strategy. Their emphasis on error convergence and image artifacts is useful for the present paper because our experiment also evaluates reconstruction quality through MSE, MAE, and visual error maps.

Qiskit provides tools for constructing circuits and studying Grover-style operators. In the present paper, those tools are used for a compact one-qubit experiment over rendering-derived RGB values rather than for full-scene quantum rendering (Javadi-Abhari et al., 2024).

Beyond ray tracing and ray marching, quantum computing has also been explored for broader visualization and image-representation tasks. Bethel et al. (2023) survey how quantum computing could reshape scientific visualization pipelines, arguing that near-term devices are unlikely to replace classical visualization outright but may accelerate specific subroutines such as pattern search and data compression. Separately, quantum image representation and compression schemes, reviewed by Deb and Pan (2024), encode pixel intensities directly into qubit amplitudes or basis states, a strategy conceptually related to the single-qubit RGB channel encoding used in this paper, though at a much larger and more general scale. Together, these more recent and diverse studies indicate that current quantum-image and quantum-visualization work remains largely exploratory and simulator-based, reinforcing the motivation for the compact, diagnostic benchmark presented here.

Table 1. Closely related work and its role in the present benchmark.

Reference	Rendering target	Quantum idea	Relevance to this paper
Towards	Ray tracing / visibility	Quantum search	Motivates selected rendering subproblems,

Quantum Ray Tracing			not full pipeline replacement (Santos et al., 2022; Lu & Lin, 2022a).
Quantum Ray Marching	Light transport simulation	Quantum random walk	Shows broader quantum-rendering theory but remains far from routine use (Santos et al., 2022; Mosier et al., 2023).
Qiskit Grover tools	Search / sampling subproblem	Oracle and reflection operators	Provides circuit vocabulary for the tested amplified estimator.
Quantum Ray Casting	Ray casting / visibility	Grover-style search	Shows an early rendering-specific use of quantum search for orthographic-camera visibility and occlusion (Alves et al., 2019).
A Framework for Quantum Ray Tracing	Ray tracing / path tracing	Dense superposed sampling and quantum counting	Provides a theoretical full-scene quantum ray tracing model; contrasts with the present small RGB ROI benchmark (Lu & Lin, 2022a).
Improved Quantum Supersampling	Sampling / anti-aliasing	Robust quantum counting	Connects quantum counting to rendering error convergence and image artifacts, motivating the shot-count analysis used here (Lu & Lin, 2023).
Quantum Computing and Visualization	Scientific visualization pipelines	Pattern search / data compression	Surveys near-term quantum acceleration for visualization subroutines rather than full pipeline replacement (Bethel et al., 2023).
Quantum Image Compression Survey	Pixel / image representation	Amplitude and basis-state qubit encoding	Reviews larger-scale quantum image encodings; conceptually related to the single-qubit RGB channel encoding used here (Deb & Pan, 2024).

In short, the most relevant related work supports using Mitsuba-derived rendering data alongside small Qiskit circuits as a compact benchmark, while current quantum rendering, visualization, and image-representation papers remain either subroutine accelerations or broader theoretical reformulations rather than practical end-to-end alternatives to classical path tracing (Alves et al., 2019; Bethel et al., 2023; Deb & Pan, 2024; Javadi-Abhari et al., 2024; Lu & Lin, 2022a, 2023; Mosier et al., 2023; Santos et al., 2022)

3 .Methodology

3.1 Hybrid Workflow Overview

The workflow has four stages. Mitsuba 3 renders a Cornell Box scene with custom interior geometry. A compact RGB region of interest is extracted from the reference image. Qiskit estimates each channel independently with one-qubit circuits. The estimated channels are reconstructed into RGB patches and compared against the reference patch.

Table 2. End-to-end workflow used in notebooks.

Stage	Tool	Output
Scene rendering	Mitsuba 3	Cornell Box low-spp and reference images
Patch extraction	Python / NumPy	8 x 8 RGB ROI values in [0, 1]
Quantum-circuit estimation	Qiskit BasicSimulator	Naive and amplified channel estimates
Evaluation	Python / Matplotlib	RGB patches, MSE/MAE metrics, error maps

3.2 Mitsuba Scene and RGB Patch Extraction

The scene uses Mitsuba 3 with the scalar variant (Javadi-Abhari et al., 2024).

The built-in Cornell Box is modified by removing the default interior objects and adding a diffuse sphere and box. The notebooks render a 256 x 256 low-sample image at 16 spp and a practical reference image at 256 spp. The reference image is not an analytical ground truth; it is a high-sample classical baseline for this compact experiment.

The selected region of interest is represented as a height-by-width RGB tensor. In the reported notebook run, H = 8 and W = 8, giving 64 pixels and 192 scalar channel values.

$$\mathrm{ROI} \in \mathbb{R}^{H \times W \times 3} \quad (1)$$

Equation (1). RGB region-of-interest tensor representation.

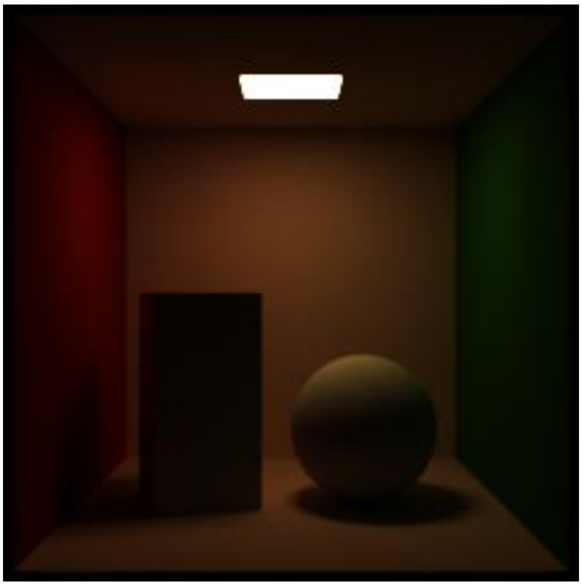

Figure 1. Reference Mitsuba Cornell Box render used as the source image for the RGB patch.

Figure 2. Reference image with selected 8 x 8 ROI, low-spp ROI, and reference ROI.

3.3 One-Qubit Amplitude Encoding

Each RGB channel value v in [0, 1] is treated as an independent scalar target. The circuit applies an Ry rotation with theta = 2 arcsin (sqrt(v)) to the initial state |0>. This prepares a qubit whose probability of measuring |1> is v. The exact statevector check in the notebook confirms this relationship, for example red-channel value: target v = 0.336877 and exact P(1) = 0.336877.

$$\theta = 2\arcsin(\sqrt{v}) \quad (2)$$

Equation (2). Rotation angle used to encode one RGB channel value.

$$P(1) = \sin^2\left(\frac{\theta}{2}\right) = v \quad (3)$$

Equation (3). Measurement probability of the prepared one-qubit state.

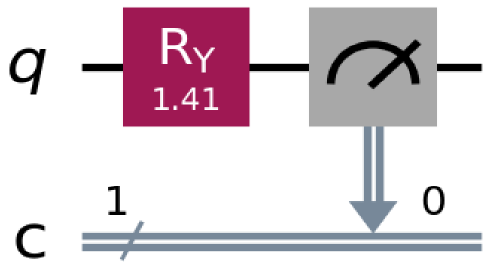

Figure 3. Naive one-qubit amplitude-encoding circuit for a single RGB channel value.

3.4 Naive Sampling Estimator

The naive estimator repeatedly measures the amplitude-encoded qubit. If n1 is the number of measurement outcomes equal to 1 and Nshots is the total number of shots, the channel estimate is v_hat = n1 / Nshots. This is Bernoulli sampling of the target probability and is applied independently to the R, G, and B channels of each pixel.

$$\hat{v}_{\text{naive}} = \frac{n_1}{N_{\text{shots}}} \quad (4)$$

Equation (4). Naive shot-based estimator for one channel value.

$$\hat{I}_{ij} = [\hat{R}_{ij}, \hat{G}_{ij}, \hat{B}_{ij}] \quad (5)$$

Equation (5). Reconstructed RGB estimate for one pixel.

3.5 Grover-Style Amplified Estimator

The amplified estimator applies a one-qubit Grover-style iterate before measurement and then attempts to invert the measured amplified probability. In ideal amplitude amplification, if v = sin^2(alpha), repeated amplification transforms the success probability according to sin^2((2k + 1) alpha), subject to branch and iteration-depth constraints.

$$Q = AS_0A^{\dagger}S_f \quad (6)$$

Equation (6). Grover-style operator form used to motivate the amplified circuit.

$$p_k = \sin^2((2k+1)\alpha), \quad v = \sin^2(\alpha) \quad (7)$$

Equation (7). Ideal amplified probability model for k iterations.

$$\hat{v}_{\mathrm{amp}} = \sin^2\left(\frac{\arcsin(\sqrt{p_k})}{2k+1}\right) \quad (8)$$

Equation (8). Probability inversion used to recover an estimated channel value.

In the current notebook implementation, this amplified estimator does not recover the original channel values correctly. For the first red-channel example, the target is 0.336877, but the amplified recovered estimate is about 0.043815 at 512 shots. The paper therefore treats the amplified result as a diagnostic negative result, not as evidence that amplitude amplification improves reconstruction.

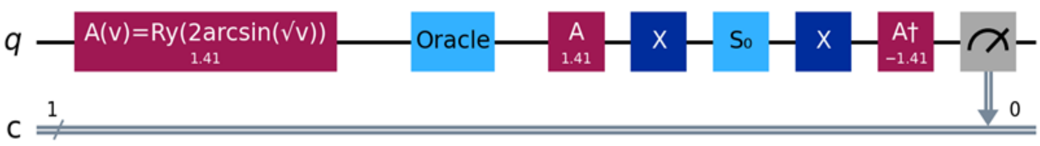

Figure 4. Grover-style amplified circuit used in the experiment. The measured behavior indicates that this implementation or its inversion requires further validation.

3.6 Evaluation Metrics

Reconstructed patches are compared against the reference RGB patch using mean squared error (MSE), mean absolute error (MAE), per-channel MSE, and visual error maps. The shot-count experiment uses 32, 64, 128, 256, 512, and 1024 shots.

$$\mathrm{MSE} = \frac{1}{3HW}\sum_c\sum_i\sum_j (I_{ijc} - \hat{I}_{ijc})^2 \quad (9)$$

Equation (9). Mean squared error over RGB patch channels.

$$\mathrm{MAE} = \frac{1}{3HW}\sum_{c}\sum_{i}\sum_{j}\left|I_{ijc} - \hat{I}_{ijc}\right| \quad (10)$$

Equation (10). Mean absolute error over RGB patch channels.

4. Experimental Setup

Table 3. Experimental configuration used by the notebooks.

Parameter	Value used
Renderer	Mitsuba 3, scalar_rgb variant
Scene	Cornell Box with custom diffuse sphere and box
Image resolution	256 x 256
Low-sample render	16 spp
Reference render	256 spp
ROI size	8 x 8 pixels, centered in the reference image
Channel values	64 pixels x 3 channels = 192 scalar targets
Qiskit backend	BasicSimulator
Shot counts	32, 64, 128, 256, 512, 1024
Amplification depth	k = 1

5 .Results

The results show a clear separation between the two estimators. The naive estimator behaves like ordinary Bernoulli sampling: increasing the number of shots reduces reconstruction error. The amplified estimator, as currently implemented, remains nearly constant at high error across all shot counts.

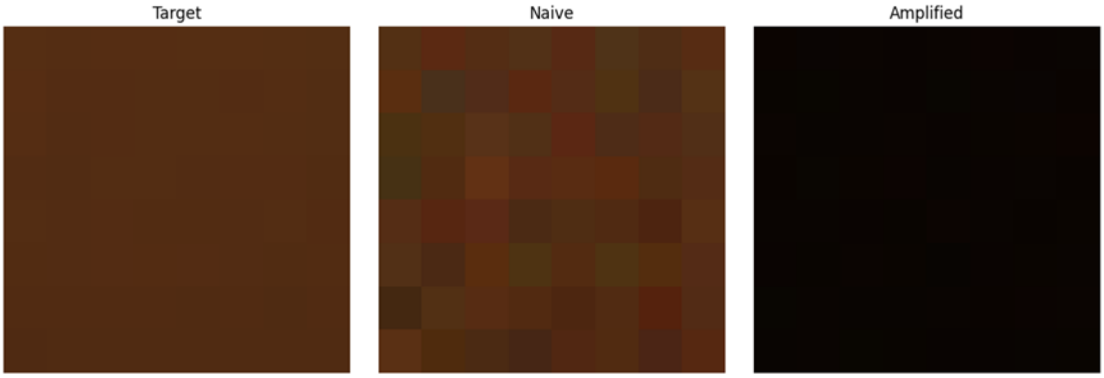

Figure 5. Target RGB patch, naive reconstruction, and amplified reconstruction at 512 shots.

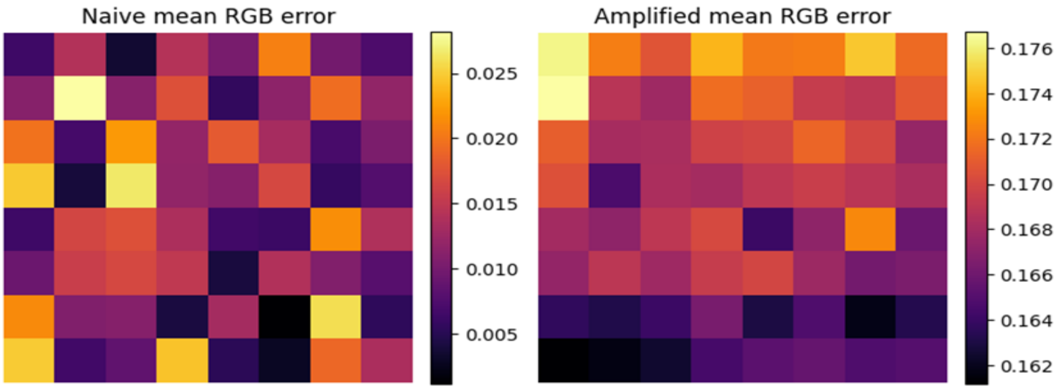

Figure 6. Mean RGB error maps for naive and amplified estimators.

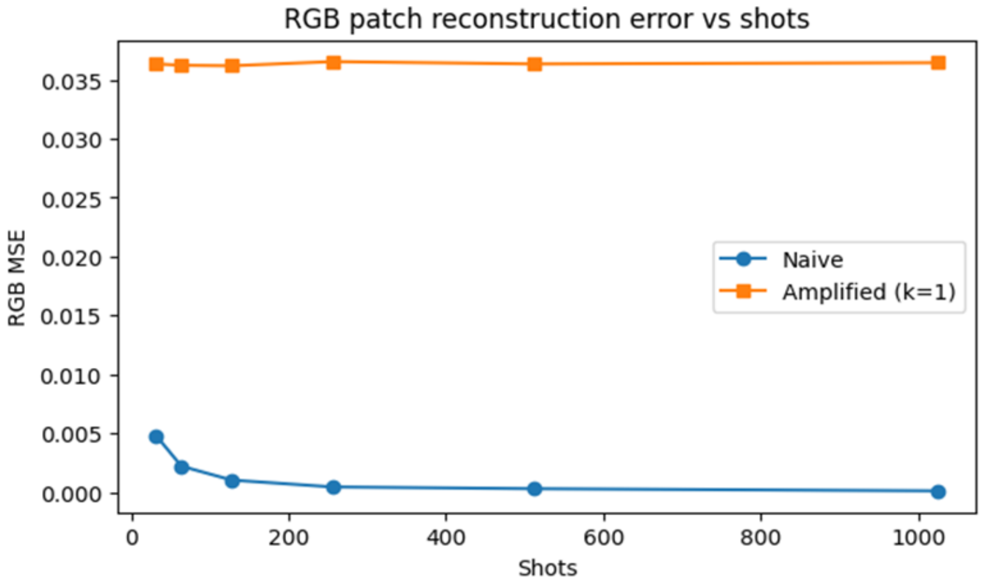

Figure 7. RGB MSE versus shot count. Naive sampling improves with shots; the tested amplified estimator remains near 0.036 MSE.

5.1 Shot-Count Metrics

Table 4. RGB reconstruction error over shot counts.

Shots	Naive MSE	Amplified MSE	Naive MAE	Amplified MAE
32	0.00480	0.03634	0.05437	0.16839
64	0.00222	0.03623	0.03766	0.16794
128	0.00104	0.03619	0.02543	0.16780
256	0.000457	0.03654	0.01698	0.16870
512	0.000307	0.03634	0.01366	0.16823
1024	0.000124	0.03645	0.00895	0.16859

5.2 Per-Channel Error

Table 5. Per-channel MSE at 512 shots.

Estimator	Red MSE	Green MSE	Blue MSE
Naive	0.000487	0.000211	0.000122
Amplified (k = 1)	0.080441	0.023922	0.004398

The per-channel table shows that the amplified estimator's error is especially severe in the red channel. This is consistent with the example-pixel failure, where a red target near 0.337 is recovered near 0.044. The amplified reconstruction therefore appears dark and biased rather than simply noisy.

6. Discussion

The experiment should be interpreted as a compact hybrid benchmark, not as a rendering speedup demonstration. Mitsuba performs all physically based rendering, while Qiskit estimates scalar RGB values extracted from an already rendered image. This distinction is important: the method does not estimate radiance integrals, visibility, light paths, or path-tracing samples.

The naive estimator validates the encoding and measurement pipeline. Its errors decrease with shot count, as expected for Bernoulli sampling. The amplified estimator does not validate the intended improvement. Because amplitude amplification is sensitive to circuit ordering, reflection definitions, iteration depth, and inverse mapping branch choices, the present result should be reported as a failure case that motivates further circuit-level checks.

This negative result is still scientifically useful. It prevents an overstated claim and identifies exactly where future work should focus verifying the one-qubit amplified transformation analytically, testing the circuit with statevectors before shot sampling, and comparing alternative amplitude-estimation procedures.

7. Limitations

The method does not perform full quantum ray tracing, full quantum path tracing, or quantum light-transport simulation.

Each RGB channel is estimated independently; spatial correlation and color-channel correlation are not encoded.

The 256 spp reference image is a practical finite-sample baseline, not analytical ground truth.

The amplified estimator currently underperforms and should not be interpreted as evidence of quantum advantage.

The experiments are simulator-based and do not measure hardware noise, queue latency, runtime cost, or practical speedup.

The current evaluation is based on one scene and one centered 8 x 8 ROI; broader tests are needed before generalizing.

8. Future Work

Validate the amplified circuit with exact statevector probabilities before applying shot-based inversion. Test branch-aware inversion and different k values for channel ranges where amplification is theoretically well conditioned. Compare against established amplitude-estimation variants rather than only a hand-built Grover-style circuit. Repeat the experiment over multiple ROIs, higher-spp references, and multiple random seeds. Explore multi-qubit encodings that can represent spatial or RGB-channel structure rather than independent scalar channels. Evaluate the workflow on IBM Runtime Sampler only after simulator behavior is analytically validated.

9. Conclusion

This paper presented a reproducible Mitsuba-Qiskit workflow for quantum-inspired RGB patch reconstruction. Mitsuba 3 rendered a Cornell Box scene and supplied an 8 x 8 RGB reference patch. Qiskit then encoded each RGB channel value into a one-qubit amplitude and estimated the values with two strategies. The naive estimator behaved as expected and improved with shot count. The tested Grover-style amplified estimator did not improve reconstruction and instead produced a persistent bias, making it a diagnostic result rather than a successful acceleration method.

The main value of the work is therefore its clear bridge between physically based rendering data and quantum-circuit estimation experiments. With corrected amplitude-amplification validation, repeated-run statistics, and broader scene/ROI tests, the workflow can serve as a useful foundation for early-stage research and teaching in quantum-inspired rendering experiments.

References

- Alves, C., Santos, L. P., & Bashford-Rogers, T. (2019). A quantum algorithm for ray casting using an orthographic camera. In *2019 International Conference on Graphics and Interaction (ICGI)* (pp. 56–63). https://doi.org/10.1109/ICGI47575.2019.8955061
- Bethel, E., Amankwah, M., Balewski, J., Van Beeumen, R., Camps, D., Huang, D., & Perciano, T. (2023). Quantum computing and visualization: A disruptive technological change ahead. *IEEE Computer Graphics and Applications*, 43(6), 101–111.
- Deb, S. K., & Pan, W. D. (2024). Quantum image compression: Fundamentals, algorithms, and advances. *Computers*, 13(8), 185.
- Huo, Y., & Yoon, S.-E. (2021). A survey on deep learning-based Monte Carlo denoising. *Computational Visual Media*, 7(2), 169–185.
- Javadi-Abhari, A., Treinish, M., Krsulich, K., Wood, C. J., Lishman, J., Gacon, J., Martiel, S., Nation, P., Bishop, L. S., Cross, A. W., Johnson, B., & Gambetta, J. (2024). *Quantum computing with Qiskit*. arXiv. https://doi.org/10.48550/arXiv.2405.08810
- Lu, X., & Lin, H. (2022a). *A framework for quantum ray tracing*. arXiv. https://doi.org/10.48550/arXiv.2203.15451
- Lu, X., & Lin, H. (2023). Improved quantum supersampling for quantum ray tracing. *Quantum Information Processing*, 22, 359. https://doi.org/10.1007/s11128-023-04114-x
- Martyn, J., Rossi, Z. M., Tan, A. K., & Chuang, I. L. (2021). Grand unification of quantum algorithms. *PRX Quantum*, 2(4), 040203.
- Mosier, L., McGuire, M., & Hachisuka, T. (2023). *Quantum ray marching: Reformulating light transport for quantum computers*. In *SIGGRAPH Asia 2023 Conference Papers*.
- Plekhanov, K., Rosenkranz, M., Fiorentini, M., & Lubasch, M. (2021). Variational quantum amplitude estimation. *arXiv*. https://doi.org/10.48550/arXiv.2109.03687

- Santos, L. P., Bashford-Rogers, T., Barbosa, J., & Navrátil, P. (2022). Towards quantum ray tracing. *arXiv*. https://doi.org/10.48550/arXiv.2204.12797
- Yan, R., Guo, H., Huang, L., Xiao, N., Li, S., Wang, Y., Lv, Y., & Chen, G. (2025). A survey on deep learning for Monte Carlo path tracing. *ACM Computing Surveys*, 58(4), 1–38.