International Conference on Emerging Trends in Technology and Management ISSN: 2349-6002

# Realtime Rendering with Ml-Enhanced Materials

Abhiram S[1], Amaan Liyah[2], Pranave Kannan[3], Sooraj S[4], Yatheesh S[5], Nikhila A[6]
[1,2,3,4,5]*Student, Sarabhai Institute of Science & Technology*
[6]*Assistant Professor, Sarabhai Institute of Science & Technology*
doi.org/10.64643/IJIRTV12I10-204463-459

***Abstract*—Real-time rendering is of paramount importance for modern applications such as gaming, virtual reality and interactive simulations. Achieving high visual realism under tight performance constraints remains a significant challenge. Common rendering pipelines based on shaders rely on manually defined material models that often lack the flexibility and accuracy required to efficiently capture complex light-material interactions. In this paper we present a novel approach of applying machine learning techniques in real time shader systems to improve material representation and rendering quality. This approach uses a trained neural network to simulate complex material properties such as reflection, displacement, and texture details, giving a realistic output while keeping real-time performance. This system consists of machine learning models integrated into the shader pipeline, which give material responses to changes in environment lighting and other scene conditions. This research shows that an ML-enhanced material system provides better output than traditional shader techniques with small computational overhead and minimal impact on frame rates.**

***Index Terms*—Computer Graphics, Deep Learning/ Machine Learning, Neural Rendering, Physically Based Rendering (PBR), Shaders**

## I. INTRODUCTION

For decades, Physically Based Rendering (PBR) has been a trademark for achieving visual enhancement in computer graphics. PBR allows for realistic reflections, shadows, and material properties by simulating the physical interaction between light and surfaces. Traditional Shader pipelines have reached the limits of its computational power which demands photorealism in applications like high-fidelity gaming, virtual reality, and digital twin. Complex optical phenomena like subsurface scattering, multi-layered cloth dynamics, and global illumination are expensive for computation [1].

For overcoming these constraints, technologies like Machine Learning (ML)- enhanced materials are used. It uses Neural Rendering for mathematical approximation of light. Instead of using mathematical computational methods it uses pre-trained models like Generative Adversarial Networks (GANs) or Multi-Layer Perceptrons (MLPs), which is used to predict complex material responses and environmental lighting effect with high precision [3].

By leveraging ML, developers can achieve "physically realistic to photorealistic" transitions that were previously only possible in offline cinema rendering. Research shows that integrating ML models—such as recurrent neural networks for sequence generation or diffusion blocks for light scattering—allows for a significant leap in visual quality with minimal computational overhead [3].

This research explores the synergy between traditional physics engines and neural inference, demonstrating how a hybrid approach can maintain real-time performance while delivering the fine-grained texture details and dynamic material responses essential for the next generation of real-time graphics.

## II. TEXTURES, SHADERS AND MATERIALS

### *A. TEXTURES*

A texture is a 2D data map (usually an image) applied to the surface of a 3D model to provide visual detail. Beyond just colors (Albedo), textures can store numerical data to tell the computer about surface height (Normal maps), shininess (Roughness maps), or transparency (Alpha maps).

### *B. SHADERS*

A shader is a small program—written in languages like HLSL or GLSL—that runs on the GPU to calculate the final color of every pixel on the screen. It takes inputs like light direction, camera position, and texture data to compute how light should bounce off a surface.

### C. MATERIALS

A material is a high-level container or "recipe" that defines the overall physical identity of an object by combining textures and shaders with specific settings. The material tells the Shader which Textures to use and provides the physical constants (like how metallic or smooth it is) to complete the calculation.

In the simplified 3d rendering workflow, textures including color map, normal map, and roughness map. In the shader, the programming executes the mathematical instructions.

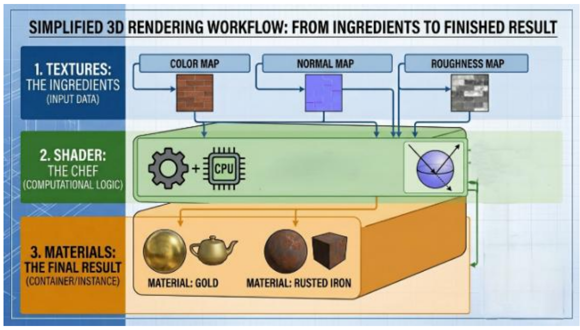

Figure 1: 3D RENDERING WORKFLOW

## III. ML-ENHANCED MATERIALS

Machine Learning (ML)-enhanced materials is an advancement in computer graphics where traditional mathematical formulas are replaced by neural networks for simulating real models. Using Machine Learning (ML)-enhanced materials, system trains complex interactions from high-quality data, producing photorealistic output.

### A. Neural Rendering & Distribution Transfer

A 3D model is displayed using GeRM (Generative Rendering Model) which generates photorealistic output. model takes basic physical attributes (like depth and color) and iteratively enriches them with details the computer didn't originally have, such as the specific way sunlight glows through a window or the way fabric naturally wrinkles on a sofa. [5]

### B. Generative Adversarial Shaders (GAS)

The method addresses the high computational cost of traditional neural network-based rendering, enabling high-quality realism enhancement on resource-constrained devices, such as embedded and mobile GPUs. The pipeline is trained using an adversarial framework (similar to GANs) to learn a mapping between raw rendered game frames and a target style (e.g., photo-realistic or a target dataset like Cityscapes). [6]

### C. Hybrid-Physics Neural Models

Neural Network (GAN or RNN) are used to predict motion and texture changes (like cloth, hair or water). While, traditional physics engines predict the basic shape, Neural Network can predict the fine details of the surface. Hybrid-physics neural models, or Physics-Informed Neural Networks (PINNs) and their hybrids, combine data-driven machine learning with known governing physical laws (e.g., Navier-Stokes, Maxwell's equations) to improve accuracy, interpretability, and generalization. They act as robust surrogate models that require less data for training by embedding physical constraints directly into the loss function.[8]

## IV. TRADITIONAL METHODS

In standard computer graphics, this is known as the Physically Based Rendering (PBR) pipeline. Instead of "learning" how to look, these methods use hard-coded mathematical formulas to simulate physics.

### A. The Mathematical Foundation (BSDF)

Traditional materials are based on the Bidirectional Scattering Distribution Function (BSDF). This is a complex math equation that tells the computer: "If light hits this point from Angle A, how much light should bounce toward the camera at Angle B?" Writing these formulas for complex surfaces like human skin, silk, or car paint is extremely difficult. Scientists have to spend years creating specific math models (like the GGX Microfacet model) just to make a surface look "mostly right."

### B. Uber-Shaders: The "One Size Fits All" Approach

Because GPUs have limited time to render a frame (usually under 16ms), traditional engines use Uber-Shaders. A single, massive piece of code that contains the math for every possible material. To make a "Gold" material, you just turn the "Metallic" slider to 1.0 and the "Roughness" slider to 0.1. This makes everything look slightly "plastic" or "samey." It’s a "jack of all trades, master of none" approach that struggles with fine details like the way light glows through a leaf (subsurface scattering).

### *C. The "Baked" Data Bottleneck*

To get high detail, traditional methods rely on Textures (images) that are "baked" or pre-calculated. Artists create high-resolution maps (Albedo, Normal, Metallic) and wrap them around 3D objects. These textures take up massive amounts of memory (VRAM). If you want a 4K texture for every rock in a game, you will quickly run out of memory.

## V. MODERN TECHNOLOGIES

### *A. DLSS 5: Intra-Frame Generative Rendering*

DLSS 5 moves beyond simply generating new frames (DLSS 3) or reconstructing rays (DLSS 3.5) by directly participating in the creation of the scene's pixels within a single frame. The AI model is trained to analyze the game's base rendering (color and motion vectors) to understand scene semantics, such as hair, fabric, skin, and environmental lighting. It uses this information to "infuse" high-fidelity details—such as sub-surface scattering on skin or realistic reflections—that would be computationally expensive to render via traditional methods. [7]

### *B. SLANG.D & Differentiable Shading*

A major bottleneck in ML-enhanced materials was the difficulty of making shaders "talk" to neural networks. SLANG.D is a new shading language with first-class automatic differentiation support. SLANG.D is an extension of the Slang shading language designed for high-performance, automatic differentiation in graphics pipelines. It enables fast, modular, and differentiable rendering, allowing complex shaders—like path tracers—to be optimized via gradient-based methods. SLANG.D generates efficient derivatives, competing with handwritten code while supporting modern features like nested differentiation and dynamic dispatch.

### *C. RTX Mega Geometry & ReSTIR PT*

NVIDIA RTX Mega Geometry and ReSTIR PT (Path Tracing) are advanced rendering technologies introduced for NVIDIA Blackwell (RTX 50-series) and later, designed to enable full-fidelity path tracing on massive, cinematic-scale scenes. Mega Geometry accelerates ray tracing of dense geometry (e.g., Nanite) by up to 100x through efficient BVH updates, while ReSTIR PT enhances indirect lighting, allowing for interactive, real-time rendering of complex environments, foliage, and lighting.

### D. CNNs (Convolutional Neural Networks)

CNNs are the workhorses for extracting features from textures. They are designed to "look" at patterns, edges, and gradients in an image. They are used to analyze raw texture data (like a photo of wood) and break it down into different maps (Normal, Roughness, Metallic). In real-time ray tracing, CNNs are used to "clean up" noisy images. Instead of the GPU calculating billions of light rays, it calculates a few, and the CNN predicts what the smooth, finished image should look like. CNNs can take a small sample of a material (like a square of carpet) and generate a massive, non-repeating version of it that looks natural.

### E. *GANs* (Generative Adversarial Networks)

GANs are the "artists" of the ML world. They consist of two networks: a Generator (which tries to create a realistic image) and a GANs are used to train shaders to match a "target" style. For example, a shader can be trained on real-world photos of a city so that it learns to add realistic "grit," puddles, and lighting reflections to a basic 3D game scene. In the GeRM framework [5], GAN-like logic is used to "evolve" a physically realistic render into a photorealistic one. It iteratively adds details like natural fabric wrinkles or the way light glows through a lamp shade.

## VI. APPLICATIONS

### 1. High-Fidelity Gaming & Esports

This is the primary driver for technologies like DLSS 5 and Generative Adversarial Shaders (GAS).

Performance Scaling:

Games can be rendered at lower internal resolutions to maintain high frame rates (144Hz+), while ML fills in the "missing" photorealistic details like skin pores or fabric weaves.

Dynamic Environments:

Using GAN-based shaders, game environments can realistically "weather" in real-time—showing puddles forming during rain or clothes becoming damp and heavy—without pre-baked animations.

Reduced Storage:
Instead of downloading 100GB of 4K textures, games can use smaller "neural textures" that the GPU expands into high-detail surfaces on-the-fly.

2. Digital Twins & Industrial Simulation
Industries use ML-enhanced materials to create perfect digital replicas of real-world objects.

Predictive Maintenance:
Hybrid physics-neural models (like the cloth simulation research by Qiu [3]) allow engineers to simulate how materials like carbon fiber or industrial fabrics will wear, stretch, or fail over time under stress.

Architecture & Urban Planning:
Tools like GeRM [1] allow architects to visualize how "warm sunlight" or specific lamp glows will interact with interior materials at different times of day with cinematic accuracy, aiding in lighting design before construction begins.

3. Virtual Production & Filmmaking
The "Volume" (LED wall) technology used in shows like The Mandalorian relies on real-time rendering.

In-Camera Visual Effects (ICVFX):
ML-enhanced materials allow the background on LED walls to react to physical light changes on set instantly. If a stage light changes color, the "Neural Materials" in the background update their reflections to match perfectly, eliminating the need for expensive post-production.

Digital Humans:
Neural rendering allows for lifelike skin and hair—the most difficult materials to render—to be manipulated in real-time for live-action broadcasts or virtual influencers.

4. E-Commerce & Virtual Try-Ons
The hybrid neural-physics approach is revolutionizing how we shop online.

Virtual Dressing Rooms:
As detailed in the research by Qiu [2025], ML allows for highly accurate, real-time cloth simulation. Shoppers can see how a specific fabric (silk vs. denim) drapes and moves on their own 3D avatar with 60 FPS fluidity.

Product Visualization:
Furniture and jewellery retailers use ML-enhanced materials to let customers see how light refracts through a diamond or reflects off a velvet sofa in their own living room via Augmented Reality (AR).

5. Medical & Scientific Visualization
Surgical Training:
Real-time shaders can simulate the translucent and reflective properties of human tissue (subsurface scattering) more accurately than traditional methods. This provides medical students with a more realistic "feel" during VR surgical simulations.

Molecular Dynamics:
Scientists use generative rendering to visualize complex protein folding or chemical reactions, where the ML can "predict" the visual surface of a molecule based on its atomic data.

## VII. CONCLUSION

The landscape of computer graphics is currently undergoing a foundational shift as traditional, formula-based rendering methods are being augmented by Machine Learning (ML) enhanced materials. In a standard rendering pipeline, Shader’s act as the computational logic, using hard-coded mathematical models like the Bidirectional Scattering Distribution Function (BSDF) to calculate how light interacts with surfaces. However, these traditional models often struggle with complex, non-linear visual phenomena such as subsurface scattering in human skin or the intricate anisotropic reflections of silk. By integrating ML, specifically Convolutional Neural Networks (CNNs) and Generative Adversarial Networks (GANs), developers can now replace or assist these heavy math calculations with neural inference. This allows the GPU to "predict" photorealistic details rather than brute-forcing them, enabling cinematic-quality visuals within the strict millisecond budgets required for real-time applications.

Modern advancements like DLSS 5 and Generative Adversarial Shaders (GAS) demonstrate the power of this hybrid approach. These technologies utilize "semantic awareness" to identify specific material types and apply context-aware enhancements—such as adding natural fabric wrinkles or realistic surface "grit"—that would be computationally prohibitive to simulate physically.

Furthermore, the rise of Differentiable Rendering and specialized shading languages like SLANG.D has made it possible to train shaders directly from real-world image data. This "Inverse Rendering" capability allows for the creation of digital twins that are indistinguishable from their physical counterparts. Ultimately, the synergy between ML and real-time shaders is moving the industry toward a generative rendering paradigm, where the GPU functions as an intelligent predictor of reality, offering a scalable path toward true photorealism in gaming, industrial simulation, and virtual production.

## ACKNOWLEDGMENT

The development and exploration of real-time shaders enhanced by machine learning owe a significant debt to the pioneering research in neural rendering and differentiable physics. We acknowledge the foundational contributions of the authors of the GeRM framework [1] and Generative Adversarial Shaders [2], whose work in distribution transfer and adversarial training has redefined the boundaries of photorealism on consumer-grade hardware. Special recognition is also due to the research teams at NVIDIA for the development of DLSS 5, which bridged the gap between deterministic graphics and generative AI. Furthermore, we express our gratitude to the contributors of the hybrid physics-ML model [3] and 3D cloth simulation studies [4], whose integration of neural networks with physical engines has provided the scalability necessary for complex material dynamics. Finally, we acknowledge the academic and open-source communities for providing the datasets and shading languages, such as Slang, that facilitate the ongoing convergence of deep learning and computer graphics.

## REFERENCES

[1] J. Lu *et al.*, “GeRM: A Generative Rendering Model from Physically Realistic to Photorealistic,” *arXiv preprint arXiv:2604.09304*, Apr. 2026. [Online]. Available: DOI arXiv 2604.09304

[2] A. Salmi, Sz. Cséfalvay, and J. Imber, “Generative Adversarial Shaders for Real-Time Realism Enhancement,” Computer Graphics Forum, vol. 42, no. 4, pp. 11–22, May 2023. [Online]. Available: DOI CGF 14870

[3] J. Qiu, “Hybrid Neural Network and Physics Engine for Real-time 3D Cloth Simulation,” Informatica, vol. 49, no. 8, pp. 161–174, 2025. [Online]. Available: DOI Informatica 6965

[4] Z. Zhang and E. Simo-Serra, “Neural Scene Baking for Permutation Invariant Transparency Rendering with Real-time Global Illumination,” NVIDIA Research Technical Report, 2026.

[5] S. A. Malik, T. A. S. Doherty, B. Colmey, S. J. Roberts, Y. Gal, and P. A. Midgley, “Hybrid Physics-Machine Learning Models for Quantitative Electron Diffraction Refinements,” *arXiv preprint arXiv:2508.05908v1*, Aug. 2025.