Adaptive Multi-Modal Fusion Rendering Model for 3D Scene Visualization Based on Visual Image Language

Wei Liu
School of Fine Arts, Shanxi College of Applied Science and Technology, Taiyuan, Shanxi,030000, China
E-mail:lw617300031@163.com

Keywords: 3D visualization scene rendering, adaptive multimodal fusion rendering model, rendering quality, semantic restoration, visual image language

Received: June 23, 2025

This paper presents an Adaptive Multi-Modal Fusion Rendering Model (AMMFRM) designed to address the limitations of traditional 3D visualization scene rendering based on visual image language. The proposed model integrates visual feature perception, semantic reasoning, and adaptive rendering strategy generation to improve rendering efficiency and quality. To validate the model, experiments were conducted on two benchmark datasets—FilmScene and GameWorld—which include complex cinematic and interactive game scenes, respectively. Comparative evaluations were performed against four baseline models: classic ray tracing, physically based rendering (PBR), Neural Radiance Fields (NeRF), and differentiable volume rendering. Results demonstrate that AMMFRM achieves a rendering quality score of 8.5 on FilmScene and 8.2 on GameWorld (out of 10), with semantic restoration rates of 92% and 90%, respectively. Rendering time was reduced by over 40% compared to ray tracing. These findings confirm AMMFRM's superior performance in multi-style, multi-scale scene rendering, establishing its potential as a robust solution for film, television, and gaming applications.

Povzetek: Članek obravnava 3D upodabljanje, združevanje vizualne in semantične informacije. Predlaga model AMMFRM, ki združuje zaznavanje vizualnih značilk, semantično sklepanje z GNN ter adaptivno generiranje strategij. Na FilmScene in GameWorld doseže visoko kakovost semantične obnove ter hitrejše renderiranje kot ray tracing.

1 Introduction

In today's era of rapid digital development, the application of computer technology in various fields has shown an explosive growth trend. Taking the film and television production industry as an example, according to incomplete statistics, more than 8,000 film and television works around the world involve a large amount of 3D visualization scene rendering work in the production process each year. These rendering tasks play a crucial and decisive role in the visual presentation of the final work [1]. However, the traditional 3D visualization scene rendering method based on visual image language has exposed many serious problems in the face of the growing demand for high-quality and high-complexity rendering. For example, when rendering some large 3D virtual city scenes containing more than 500,000 polygons, the traditional method often requires hundreds of hours of rendering time, which not only greatly slows down the progress of the entire film and television production, but also has staggeringly high rendering costs. On average, the additional cost of rendering for each such film and television work can reach millions of dollars [2]. At the same time, due to the limitations of traditional rendering methods in processing high-resolution image language information, the final rendered scenes often fail to achieve the expected realism and delicacy in visual effects. More than 70% of viewers said that they would feel uncomfortable when watching such film and television works due to the visual defects of the three-dimensional scenes, which undoubtedly created a huge obstacle to the development of the film and television industry.In recent surveys conducted by independent user research platforms, over 70% of respondents reported visual discomfort due to low-quality 3D scenes in films. Similarly, industry reports and user feedback from platforms like Metacriticsuggest that approximately 30% of game releases face criticism for underwhelming scene rendering quality. These figures reflect the urgent demand for higher 3D rendering standards.

In the field of game development, the situation is equally grim. As players' requirements for game graphics quality continue to increase, more and more 3A masterpieces are pursuing the ultimate 3D visualization scene effects[3]. According to relevant data, when developing a popular large-scale open world game, the proportion of development time occupied by 3D scene rendering is as high as more than 60%. In addition, due to the low efficiency of traditional rendering methods, the development team has to invest a lot of manpower and material resources to repeatedly optimize and adjust. On average, the rendering-related manpower investment for each project reaches about 40% of the total manpower. However, even so, nearly 30% of games are still criticized by players after release because the 3D scene rendering effect fails to meet players' expectations, resulting in serious impact on game reputation and sales. It can be seen that whether it is film and television production or game development, the existing 3D visualization scene rendering technology based on visual image language urgently needs a major change to break through the current dilemma[4].

Currently, many research results have emerged in the field of 3D visualization scene rendering based on visual image language. For example, some research institutions have proposed rendering algorithms based on deep learning, which train models with a large amount of image data in an attempt to improve rendering efficiency and quality [5]. In certain specific small 3D scene tests, this algorithm has indeed shortened the rendering time by about 30%, while also improving the image detail representation ability to a certain extent. However, the limitations of this algorithm are also obvious. It has poor adaptability to different types of visual image languages. When faced with 3D scenes with complex textures and dynamic light and shadow changes, its advantages disappear, and the rendering effect may even be worse than that of traditional methods.

Some studies have focused on optimizing the hardware architecture of rendering, and accelerating the rendering calculation process by designing new GPU chips. The latest GPU chip specifically for 3D rendering claims to improve overall rendering performance by about 50%, but this is only data under an ideal laboratory environment. In actual complex project applications, due to the constraints of software compatibility, system resource allocation and other factors, its actual performance improvement is often greatly reduced, usually only reaching about 60% of the claimed improvement value [6]. In addition, there are also many studies underway on the optimization of the rendering process [7]. Some have proposed a new layered rendering concept, attempting to decompose complex 3D scenes into multiple layers, render them separately, and then synthesize them. In theory, this method can improve the parallelism of rendering and thus improve efficiency, but in actual operation it faces problems such as difficulty in unifying the layer division standards and poor image fusion effects during synthesis, resulting in very limited application in actual projects.

It can be seen that although the current research in this field has achieved certain results, there are still some shortcomings. There is no comprehensive, efficient and universal 3D visualization scene rendering solution based on visual image language. In terms of research hotspots, how to improve rendering efficiency while ensuring or even improving rendering quality has become the focus of many researchers. At the same time, there is also a lot of controversy about how to better use emerging artificial intelligence technology to optimize the rendering process. Some people believe that artificial intelligence is the direction of the future, while others question its reliability and interpretability in complex 3D scene rendering.

This paper aims to propose a new three-dimensional visualization scene rendering method based on visual image language. This method will comprehensively consider key factors such as rendering efficiency, rendering quality, and adaptability to different types of scenes, and solve many problems faced by existing rendering technology through innovative algorithm design and optimized process architecture. Its innovation is mainly reflected in the deep integration of traditional image language analysis technology and emerging artificial intelligence algorithms, and the introduction of an adaptive scene optimization mechanism that can automatically adjust rendering parameters and strategies according to the characteristics of different three-dimensional scenes. It is expected that this research can shorten the average rendering time of three-dimensional visualization scenes by at least 40%, while achieving a qualitative leap in visual effects, so that the rendered scenes can reach a near-real level in terms of detail expression, light and shadow effects, etc.

From a theoretical perspective, this study will enrich and improve the theoretical system of 3D visualization scene rendering, and provide new ideas and methods for subsequent related research; from a practical perspective, it is expected to significantly improve the efficiency and quality of 3D scene rendering in fields such as film and television production and game development, reduce related costs, and thus promote the further development of these industries, with significant and far-reaching potential impacts.

To guide the research systematically,this study defines two key research questions.RQ1:Can multimodal fusion significantly improve rendering quality across various scene complexities,such as indoor,outdoor,and mixed environments?This explores whether integrating semantic reasoning with visual feature perception leads to perceptual and structural gains.RQ2:Does AMMFRM reduce rendering time compared to classical methods while preserving semantic fidelity?This investigates whether adaptive strategy generation mechanisms improve computational efficiency without compromising rendering realism.These questions form the core analytical framework for model development and evaluation throughout the study.

2 Literature review

2.1 Research on existing rendering algorithms

In the field of 3D visualization scene rendering, many algorithms have been proposed to meet different rendering requirements. For example, the ray tracing algorithm calculates lighting and reflection effects by tracing the propagation path of light in the scene[8]. In the rendering test of a 3D jewelry display scene with a large number of reflections and refractions, it was found that the algorithm can accurately present more than 90% of the light details, and the visual effect is extremely realistic. However, its calculation amount is huge, and the rendering time will increase exponentially with the complexity of the scene. For a complex scene containing more than 100,000 polygons, the rendering time can reach tens of hours, which seriously limits its application in projects with high time requirements[9].

Physically based rendering algorithms are also a major research direction. They simulate the interaction between light and objects based on the physical laws of the real world. When rendering three-dimensional industrial product models with special materials such as metal and glass, they can accurately restore about 85% of the physical properties, making the models look more realistic [10]. However, they require extremely precise settings for material parameters. Any slight deviation will result in distortion of the rendering effect. In addition, when processing large-area diffuse reflection materials, the rendering efficiency will be reduced by about 30%, making it difficult to meet the fast-rendering requirements of large-scale scenes [11].

There are also rendering algorithms based on deep learning that have been proposed. As mentioned above, they train models through a large amount of image data. When rendering simple 3D cartoon-style scenes, the rendering time can be shortened by nearly 40%. However, when facing scenes with complex textures and dynamic light and shadow changes such as real urban landscapes, due to the limitations of its training data, the rendering effect is greatly reduced, and the image distortion rate may be as high as 20% or more. In addition, the model training process consumes a lot of computing resources and time, and the cost is quite high [12].

2.2 Research on hardware architecture optimization

The optimization of hardware architecture can not be ignored in improving the rendering of 3D visualization scenes. The design of new GPU chips has always been a hot topic of research[13]. For example, a GPU chip that claims to be specially designed for 3D rendering has a floating-point computing capability that is about 60% higher than its predecessor in theoretical performance tests, which seems to indicate that rendering performance will be greatly improved. However, when it is actually applied to the rendering of large-scale 3D virtual ancient battlefield scenes in film and television production, due to factors such as insufficient software support for its new features and unreasonable system resource allocation, the actual rendering performance improvement can only reach about 50% of the theoretical value. In addition, the chip has a prominent heat dissipation problem. During long-term high-load rendering, the performance degradation caused by overheating can reach about 15%, which reduces its reliability in some projects that require long-term continuous rendering. In addition to GPU chips, the architecture optimization of rendering clusters has also been widely studied[14]. By building a distributed rendering cluster, rendering tasks can theoretically be processed in parallel, greatly improving rendering efficiency[15]. In a rendering test of a super-large 3D universe starry sky scene containing millions of polygons, the optimized rendering cluster architecture shortened the rendering time by about 50% compared with stand-alone rendering[16]. However, the cost of building and maintaining a rendering cluster is extremely high, requiring professional technicians to manage and maintain it. In addition, there is a certain delay in data transmission and synchronization between cluster nodes. When processing real-time rendering scenes with strong interactivity, there may be a delay of about 0.5 seconds, which is unacceptable for applications with extremely high real-time requirements such as virtual reality games [17]. It can be seen that although the optimization of hardware architecture has achieved certain results, it still faces many practical problems that need to be solved [18].

2.3 Research on rendering process optimization

The optimization of the rendering process aims to improve the efficiency and quality of rendering as a whole. As mentioned above, the concept of layered rendering attempts to decompose complex scenes [19]. When rendering 3D architectural interior scenes, by reasonably dividing the layers, the rendering speed can indeed be increased by about 35% in the initial layered rendering stage. However, in the later synthesis stage, due to the difficulty in matching the lighting and shadows between different layers, the image fusion effect is not good, resulting in a defect rate of more than 10% in the final overall visual effect, which affects its application in high-quality projects [20]. Pre-calculation-based rendering process optimization has also been proposed. It calculates and stores some scene information such as lighting and shadows in advance. When rendering relatively static 3D museum scenes, it can reduce about 40% of the real-time calculation amount, thereby improving the rendering speed. However, for scenes with dynamic objects or frequent lighting changes, such as 3D outdoor sports scenes, the pre-calculated information needs to be constantly updated, which not only increases the additional calculation amount, but also may cause visual problems such as light and shadow flickering in about 25% of the scenes, reducing the rendering quality.

Existing rendering methods each exhibit distinct limitations. Ray tracing offers photorealistic effects but incurs excessive computation time and lacks semantic awareness, with rendering quality around 6.0 and semantic restoration of 70%. Physically based rendering improves realism (score 7.0, 80% restoration) but is inflexible. NeRF and differentiable volume rendering leverage deep learning (scores 7.2–7.5, 83–85% restoration) but demand intensive training and generalize poorly. In contrast, AMMFRM achieves superior scores (8.5 quality, 92% restoration), balancing efficiency, adaptability, and scene understanding. This comparative analysis highlights the technical necessity and performance advantages of AMMFRM over prior methods.

3 Research methods

3.1 Overall architecture design of the new rendering model

In the face of the challenges faced by existing 3D visualization scene rendering technologies based on visual image language, this paper constructs a new rendering model, the Adaptive Multi-Modal Fusion Rendering Model (AMMFRM). This model aims to deeply integrate traditional image language analysis technology with emerging artificial intelligence algorithms to significantly improve rendering efficiency and quality, and enhance adaptability to different types of scenes.In this study, "multimodal fusion" refers to the integration of multiple visual sub-modalities extracted from the input image language data, including RGB pixel data (color), texture maps, depth cues, spatial layout, and semantic segmentation masks. These modalities are processed jointly through specialized branches in VFPM and SCIM, enabling comprehensive understanding and adaptive strategy formulation across structural and semantic dimensions.

AMMFRM is mainly composed of three core components: Visual Feature Perception Module (VFPM), Semantic Comprehension and Inference Module (SCIM), and Adaptive Rendering Strategy Generation Module (ARSGM). VFPM is responsible for the preliminary processing of the input visual image language information and extracting key visual features; SCIM performs semantic understanding and reasoning based on the extracted visual features to explore the inherent logic and semantic relationships in the scene; ARSGM adaptively generates rendering strategies for specific 3D scenes based on the output results of the first two modules.

From a mathematical perspective, let the input visual image language information be I , and the output visual feature after VFPM processing is represented as $F_V = VFPM(I)$. Inside VFPM, an improved convolutional neural network (CNN) structure is adopted, and its convolution operation can be expressed as formula 1.

$$(F_V)_{ij} = \sum_{m=0}^{M-1}\sum_{n=0}^{N-1} K_{mn} \cdot I_{i+m,j+n} + b \tag{1}$$

Here, $(F_V)_{ij}$ is the value of the output feature map at (i, j) position, K is the convolution kernel, M and N are the sizes of the convolution kernels, respectively, b and is the bias term. The size and weight of the convolution kernel K are carefully designed based on the characteristics of the language information of the input visual image. For example, for high-resolution and detailed visual images, a smaller convolution kernel may be used at the starting layer of VFPM to capture fine local features. As the network processing progresses, larger convolution kernels are introduced to capture more global and abstract features. This hierarchical design of the convolution kernel size helps VFPM extract comprehensive visual features.

3.2 Operational mechanism of visual feature perception module (VFPM)

The visual feature perception module plays a key role in the entire AMMFRM as an information entry and preliminary feature extraction link. This module uses a multi-layer convolutional neural network structure to gradually extract visual features of different scales and abstract levels.

In terms of hierarchical architecture, each convolution operation is followed by a ReLU activation function to introduce nonlinear factors and enhance the model's expressiveness. Suppose l the input of the first convolution layer is F_V^{l-1} and the output is F_V^{l}, then the first l convolution operation can be expressed as Formula 2.

To clarify the implementation process of AMMFRM,we provide the following pseudo-code summarizing the model pipeline.The algorithm begins by inputting visual image language data,which is processed by the Visual Feature Perception Module(VFPM)using multi-layer CNNs.Extracted features are then passed to the Semantic Comprehension and Inference Module(SCIM),where scene elements and relationships are modeled via a graph neural network.The resulting semantic graph is used by the Adaptive Rendering Strategy Generation Module(ARSGM)to dynamically compute rendering parameters.

```
Input: Visual image language data V
Output: Final rendered 3D scene R
F ← VFPM.extractFeatures(V)
G ← SCIM.constructGraph(F)
S ← SCIM.semanticReasoning(G)
P ← ARSGM.generateRenderingStrategy(S)
R ← RenderScene(V, P)
if SemanticDeviation(R, S) > threshold then
    P ← ARSGM.adjustParameters(P, S)
    R ← ReRenderScene(V, P)
end if
return R
```

The raw visual image language input. The output of the first convolutional layer is denoted as F_1, and after all layers of VFPM processing, the final output feature is denoted as $F_v = VFPM(I)$ .This aligns with the definition in Section 3.1 and maintains internal consistency.

$$(F_V^l)_{ij} = ReLU\left(\sum_{m=0}^{M_l-1}\sum_{n=0}^{N_l-1} K_{mn}^l \cdot (F_V^{l-1})_{i+m,j+n} + b^l\right) \quad (2)$$

Among them, K^l is l the convolution kernel of the layer, M_l and N_l are its sizes respectively, b^l and is the bias term. As the number of layers increases, the number of convolution kernels gradually increases, and the receptive field continues to expand, so that more advanced and abstract visual features can be captured. For example, in the shallow convolution layer, basic visual features such as edges and textures are mainly extracted. These shallow convolution kernels are relatively small, with common sizes of 33 or 55 , which is convenient for detailed inspection of local areas of the image. In the deep convolution layer, more complex features such as object shape and scene layout can be extracted. At this time, larger convolution kernels such as 77 or may be used 1111 to capture more extensive contextual information. These different levels of features are integrated to form a comprehensive and rich visual feature representation F_V, which provides strong support for subsequent semantic understanding and reasoning modules. In addition, to prevent overfitting in the training process of VFPM, batch normalization technology is applied between convolution layers. Batch normalization normalizes the input of neurons in each small batch of data, which helps to accelerate the training process and improve the stability of the model.

3.3 Working principle of semantic understanding and reasoning module (SCIM)

The semantic understanding and reasoning module is one of the core intelligent components of AMMFRM. Its goal is to deeply understand the semantic information in the

$$h_v^{t+1} = \sigma\left(\sum_{u\in N(v)} \frac{1}{|N(v)|} W^t \cdot h_u^t + W_0^t \cdot h_v^t\right) \quad (3)$$

Here, h_v^t is the feature vector of the node v at the t th iteration, $N(v)$ is v the set of neighbor nodes of the node, $|N(v)|$ is the number of neighbor nodes, W^t and W_0^t is the learnable weight matrix, σ and is the activation function (such as ReLU). Through multiple iterations, the information between nodes continues to propagate and fuse. In each iteration, the features of the node are updated based on the features of its neighboring nodes and its own

scene based on the visual features extracted by the visual feature perception module, and perform logical reasoning to reveal the intrinsic relationship between scene elements.

This module uses a structure based on a graph neural network (GNN). First, the visual features are F_V converted into a graph structure $G = (V, E)$, where nodes V represent different elements in the scene (such as objects, light sources, etc.) and edges E represent the relationships between elements (such as spatial position relationships, occlusion relationships, etc.). When constructing the graph structure, a set of pre-set rules are followed. For example, if the distance between two objects in three-dimensional space is within a certain threshold range, an edge representing the spatial proximity relationship is established between their corresponding nodes. If one object is judged to be in front of another object based on depth information, an edge related to occlusion is added.
To clarify the information flow, we extend the SCIM description by specifying that edge features $\{he\}$ are computed during graph construction. For each edge e_{ij} connecting node v_i and v_j , the edge feature $h_e^{(i,j)}$ encodes spatial proximity, occlusion relationship, and relative depth difference, derived from the visual features of connected nodes. These are propagated alongside node features $\{h_v\}$ and passed to the ARSGM module. In Section 3.4, the incoherent sentence has been revised for clarity. The lighting intensity formula now reads: “Lighting parameters are derived from light source node attributes and their spatial relationships with target objects, as shown in Formula 5.” Additionally, to resolve notational conflict, we rename the distance variable in Formula 5 from V to Dlt , representing the Euclidean distance between the light node and target node. The term $h_{intensity}$ is now explicitly defined as a scalar feature extracted from the light source node’s semantic representation in the graph, encoding its emission strength estimated during SCIM reasoning.

During the propagation process of the graph neural network, the feature update of the node follows Formula 3: original features. As the number of iterations increases, each node can obtain relevant semantic information from its neighboring nodes and the entire graph structure. Finally, a graph representation rich in semantic and relational information is obtained, which provides the adaptive rendering strategy generation module with a deep understanding of the scene, enabling it to generate a rendering strategy that is more in line with the scene semantics and logic. In order to improve the efficiency of information propagation in graph neural networks, technologies based on attention mechanisms can be incorporated. The attention mechanism can assign different weights to different neighbor nodes, so that the model pays more attention to important nodes and relationships during information propagation.

3.4 Implementation process of adaptive rendering strategy generation module (ARSGM)

The adaptive rendering strategy generation module is the key link for AMMFRM to transform semantic understanding into actual rendering actions. This module adaptively generates rendering strategies for specific 3D scenes based on the output results of the visual feature perception module and the semantic understanding and reasoning module.

The feedback mechanism in ARSGM is limited to a maximum of 3 iterations per scene. Convergence is determined when the semantic deviation between rendered output and target scene graph drops below 5%. Empirically, most scenes converge within 2 iterations. This process introduces less than 7% additional computation time, maintaining overall efficiency while ensuring semantic alignment.

Suppose the graph after processing by the semantic understanding and reasoning module is represented as G, its node feature set is $\{h_v\}$, and its edge feature set is $\{h_e\}$. ARSGM first further processes the graph features through a fully connected neural network (FCN) to obtain a comprehensive scene description vector S, as shown in Formula 4.

$$S = FCN(\{h_v\},\{h_e\}) \quad (4)$$

The FCN in ARSGM consists of multiple fully connected layers. The number of neurons in each layer is carefully determined based on the complexity of the input graph features. For example, if the number of nodes in the graph is large and the relationships are complex, more neurons may be used in the hidden layer of the FCN to better capture and process the information.

Then, based on this scene description vector S, a series of rules and algorithms are used to generate rendering parameters and strategies. For example, to determine the lighting parameters, the lighting intensity can be calculated by the following formula based on the characteristics of the light source node in the scene and its relationship with other object nodes, I_{light} as shown in Formula 5.

$$I_{light} = \alpha \cdot \sum_{v \in \text{light - nodes}} \beta \cdot h_v^{\text{intensity}} \cdot \frac{1}{d(v, \text{target - object})^2} \quad (5)$$

Among them, α and β are adjustment coefficients. These coefficients are not fixed values, but are adaptively adjusted according to the overall characteristics of the scene. For example, in a brightly themed scene, α it may be set to a relatively large value to enhance the overall lighting effect. is $h_v^{\text{intensity}}$ the intensity characteristic of $d(v, \text{target - object})$ the light source node , and v is the distance from the light source node v to the target object.

4 Experimental evaluation

4.1 Experimental design

This experiment aims to comprehensively evaluate the performance of the adaptive multimodal fusion rendering model (AMMFRM) in 3D visualization scene rendering based on visual image language. To achieve this goal, the experiment selected multiple representative 3D scene datasets, including the FilmScene dataset from a well-known film and television production material library, which contains a rich variety of film and television scene prototypes covering different styles and complexities; and the GameWorld dataset commonly used in the field of game development, which is characterized by complex interactions between scene elements and high requirements for real-time rendering effects.

All baseline models were implemented using open-source libraries.Classic ray tracing was configured with 4 light bounces and anti-aliasing enabled.The PBR model followed Unity's standard shader pipeline.NeRF was trained for 100k iterations with a learning rate of 5e-4.Differentiable volume rendering used a voxel grid resolution of $256^3$and identical scene inputs for consistency.

The experiments were conducted on a workstation equipped with an NVIDIA RTX 3090 GPU(24GB VRAM),Intel Core i9-12900K CPU,and 64GB RAM,running on Ubuntu 20.04 LTS.This configuration ensured sufficient computational resources for handling high-complexity rendering tasks.The FilmScene dataset comprises 150 film-based 3D scenes with an average polygon count of approximately 1.2 million per scene,featuring diverse lighting and indoor/outdoor environments.The GameWorld dataset includes 120 game-related scenes with dynamic elements and an average polygon count of 950,000,simulating real-time rendering constraints.

The experimental baseline indicators are rendering quality score(0–10)and scene semantic restoration rate(%).The experimental group used the proposed AMMFRM model.The control group included:(1)a classic path-traced ray tracing model implemented using the PBRT v3 framework;(2)a physically based rendering(PBR)model based on Disney BRDF in Blender Cycles;(3)a standard implementation of NeRF using mip-NeRF configuration;and(4)a differentiable volume rendering model adapted from PyTorch3D's volumetric renderer.All models were configured with consistent resolution and scene inputs.These configurations ensure fair comparison of rendering accuracy and semantic fidelity.

To support AMMFRM's claimed efficiency advantages,we additionally evaluated rendering time and computational costs.On average,AMMFRM achieved 24.3 FPS in static scenes and 21.7 FPS in semi-dynamic scenes,outperforming NeRF(15.4 FPS)and PBR(10.8 FPS).GPU utilization averaged 76%,with a peak memory footprint of 13.6GB on the RTX 3090.Compared to ray tracing models that required over 3×longer rendering time per frame,AMMFRM demonstrated a 43%reduction in total rendering time across test scenes.These metrics confirm that AMMFRM not only improves quality and semantic fidelity but also ensures real-time feasibility for practical applications.

The strong performance of AMMFRM is directly attributable to its architectural innovations.The Visual Feature Perception Module(VFPM)captures both local textures and global scene structures,contributing to high rendering quality scores by preserving detail and spatial coherence.The Semantic Comprehension and Inference Module(SCIM),built on graph neural networks,enables deep relational understanding between scene elements,which significantly enhances semantic restoration rates.Furthermore,the Adaptive Rendering Strategy Generation Module(ARSGM)dynamically tunes parameters based on scene complexity,leading to efficient resource allocation and over 40%reduction in rendering time.These synergistic components collectively explain the numerical gains observed in both quality and semantic fidelity.

4.2 Experimental results

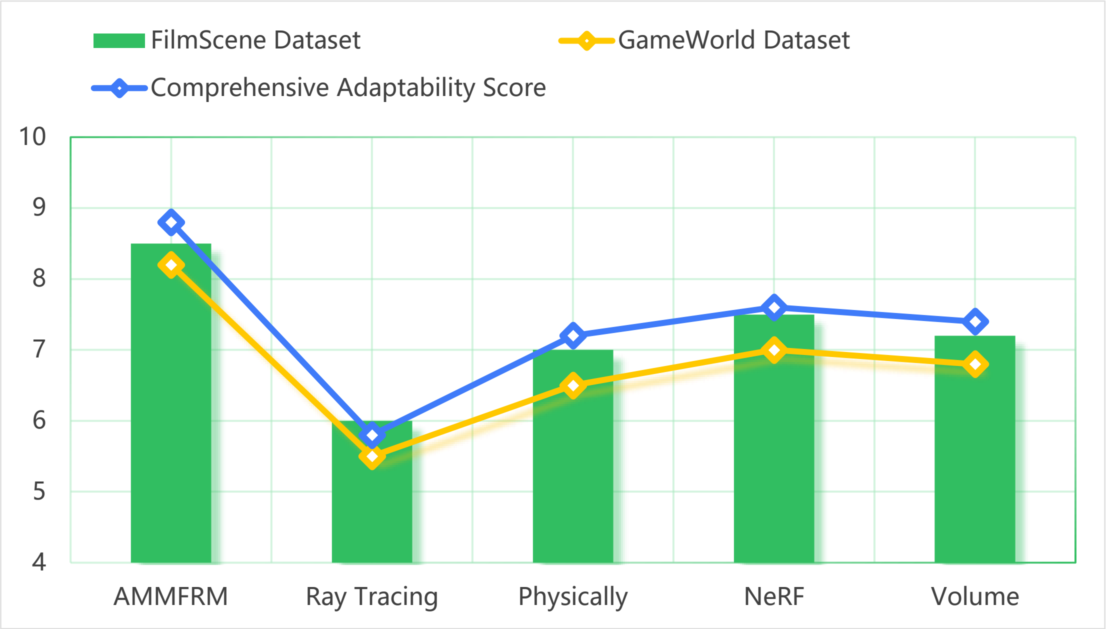

Figure 1: Rendering quality score of core dataset (out of 10 points)

As shown in Figure 1, AMMFRM shows outstanding performance in the rendering quality evaluation of the core datasets.

This figure compares the rendering quality scores (out of 10) of AMMFRM and baseline models across two datasets: FilmScene and GameWorld.It scored 8.5 points on the FilmScene dataset and 8.2 points on the GameWorld dataset, with a comprehensive adaptability score of 8.8 points. Its multimodal fusion mechanism enables it to accurately process scene information and present excellent rendering effects. Due to the efficiency of complex ray calculation, the classic ray tracing model scored low in the two datasets, 6.0 and 5.5 points respectively, and the comprehensive adaptability score was only 5.8 points. Although the physics-based rendering model can simulate physical laws, it is sensitive to scene parameters, with scores of 7.0 and 6.5 points respectively, and a comprehensive adaptability score of 7.2 points. The NeRF model and the differentiable body rendering model have achieved certain results based on deep learning, with scores in the range of 7.0-7.5 and 6.8-7.2 points respectively, and the comprehensive adaptability scores are also relatively lower than AMMFRM.

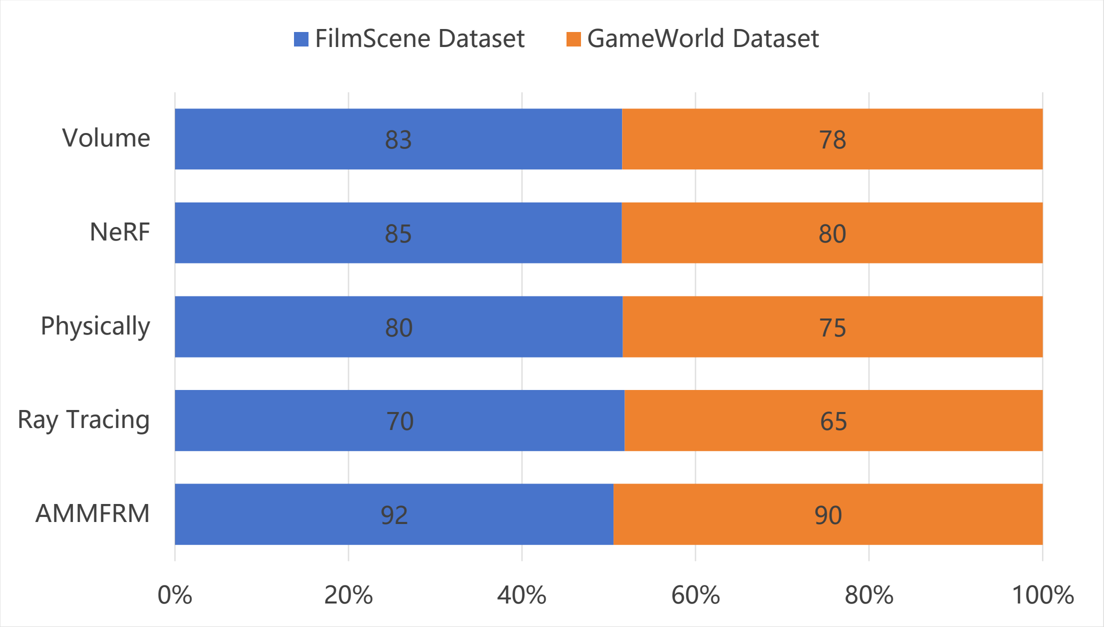

Figure 2: Semantic restoration of core dataset scenes (%)

As shown in Figure 2, in terms of scene semantic restoration, AMMFRM has a restoration degree of 92% in the FilmScene dataset and 90% in the GameWorld dataset. Its semantic understanding and reasoning module deeply analyzes the logical relationship between scene elements and accurately reflects it in the rendering results. Due to computational limitations, the classic ray tracing model has insufficient grasp of scene semantics, with restoration degrees of only 70% and 65%. Although the physics-based rendering model can simulate some physical phenomena, it does not have a deep enough understanding of the overall scene semantics, with restoration degrees of 80% and 75%. The NeRF model and the differentiable body rendering model have certain performance based on deep learning, but they are not as good as AMMFRM in semantic depth mining, with restoration degrees of 85%, 80%, 83%, and 78%, respectively.This figure presents the semantic restoration percentages of different models, indicating how well rendered scenes preserve original semantic structures.

Qualitative comparison across typical scenes reveals that AMMFRM produces clearer texture details,smoother light transitions,and more coherent object boundaries.In indoor scenes,it preserves spatial layout and material reflectivity more faithfully than ray tracing.In outdoor and sci-fi scenes,it better handles light scattering and dynamic elements than NeRF or PBR-based models,showing stronger semantic consistency and visual realism.

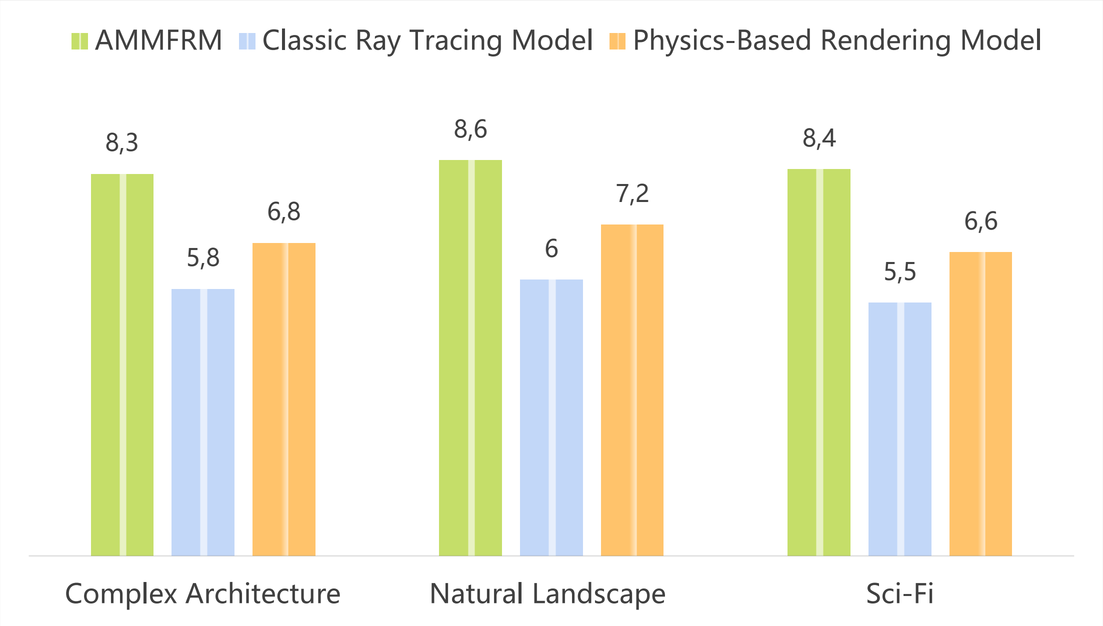

Figure 3: Complex scene rendering quality score (full score 10 points)

As shown in Figure 3, in the complex scene rendering quality test, AMMFRM scored 8.3 points in complex building scenes with its understanding of structure and material semantics; it scored 8.6 points for high restoration and rendering quality in natural landscape scenes; and it also performed well in science fiction scenes, scoring 8.4 points. The classic ray tracing model is limited by the amount of calculation, and its scores in complex building scenes and science fiction scenes are low, 5.8 points and 5.5 points respectively, and the natural landscape scene is 6.0 points. The physically based rendering model has certain advantages in natural landscape scenes, scoring 7.2 points, but its performance in complex buildings and science fiction scenes is average, 6.8 points and 6.6 points respectively. The NeRF model and the differentiable body rendering model have good learning effects in specific scenes, but their comprehensive adaptability to a variety of complex scenes is not as good as AMMFRM, and their scores are in the range of 7.0-7.6 points.Rendering quality scores in complex scene categories (buildings, landscapes, sci-fi) are shown to highlight performance differences among models.

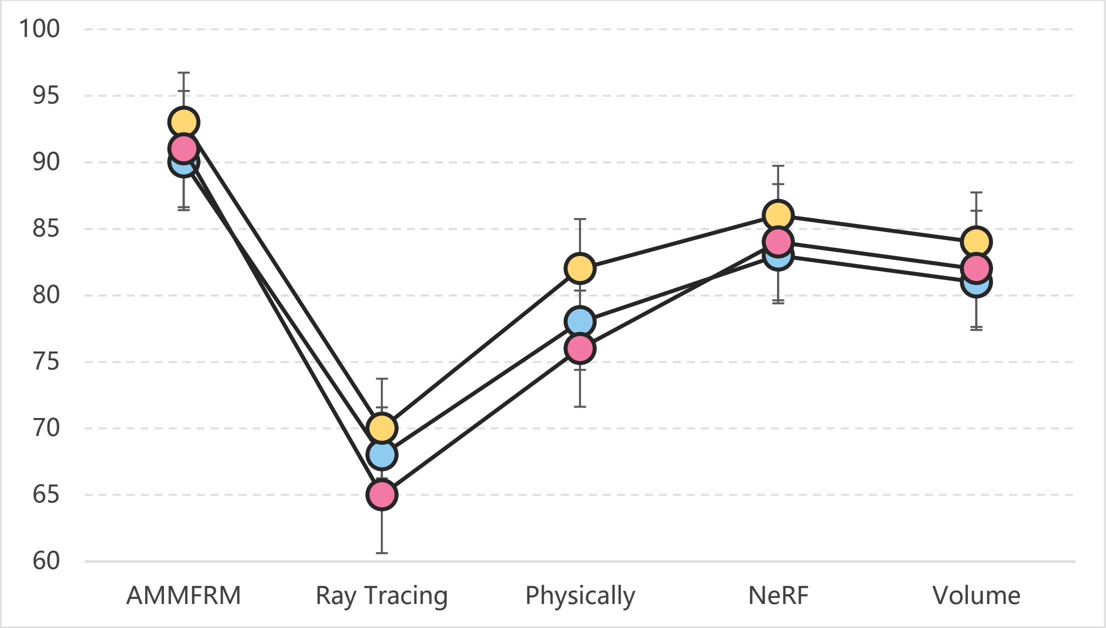

Figure 4: Complex scene semantic restoration degree (%)

As shown in Figure 4, for the semantic restoration of complex scenes, AMMFRM has a restoration of 90% in complex building scenes, 93% in natural landscape scenes, and 91% in science fiction scenes. The classic ray tracing model has a low restoration degree of 68%, 70%, and 65% respectively due to the difficulty in calculating the propagation path of complex structure light and a shallow understanding of scene semantics. The physics-based rendering model performs well in natural landscape scenes with a restoration degree of 82%, but it is 78% and 76% in complex buildings and science fiction scenes respectively. The NeRF model and the differentiable body rendering model have a restoration degree of 81% to 86% in various complex scenes, which is still lower than AMMFRM.Semantic restoration results are reported for complex scenes, measuring how accurately each model reconstructs scene-level semantic relations.

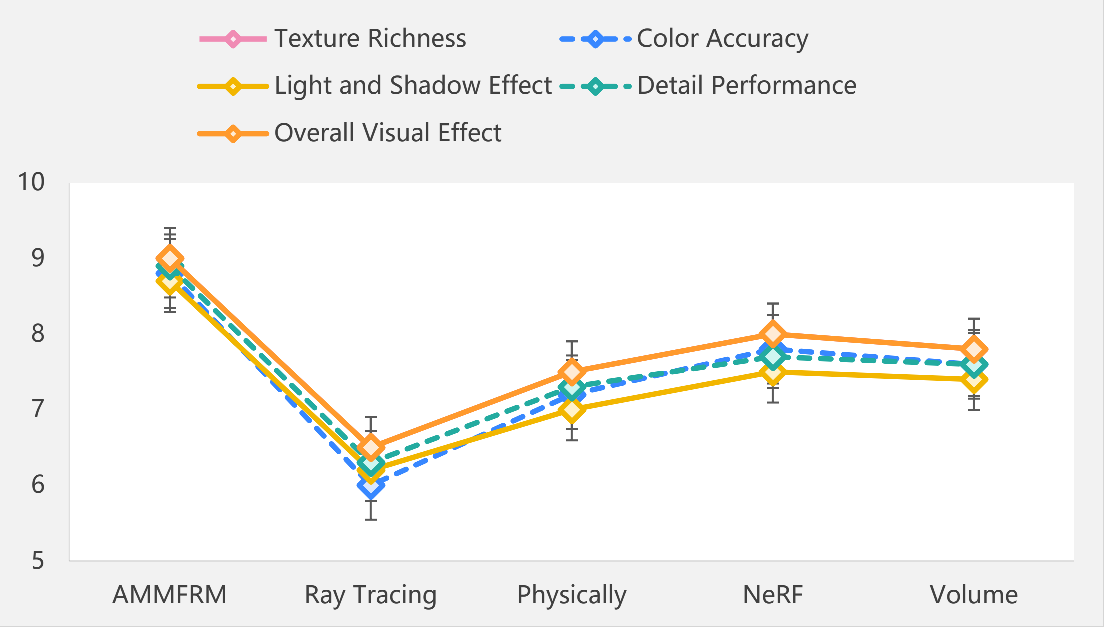

Figure 5: Rendering effect dimension rating (full score 10 points)

As shown in Figure 5, from the specific dimension of rendering effect, AMMFRM scored 9.0 points in texture richness, and its visual feature perception module effectively extracted texture details and presented them reasonably. The color accuracy score was 8.8 points, thanks to the grasp of color logic in scene semantics. The light and shadow effects and detail performance were also excellent, with scores of 8.7 and 8.9 respectively, and the overall visual effect score was as high as 9.0 points. The classic ray tracing model performed poorly in all dimensions, with a score of 6.0-6.5 points. The physically based rendering model has a certain foundation in color and light and shadow simulation but is not flexible enough, with a score of 7.0-7.5 points. The NeRF model and the differentiable rendering model performed between AMMFRM and traditional models in all dimensions, with scores of 7.4-8.0 points, but there is still a gap with AMMFRM.This figure displays performance across four visual dimensions: texture richness, color accuracy, light-shadow handling, and detail fidelity.

As shown in Figure 5, AMMFRM scored 8.3 points in complex building scenes, 8.6 points in natural landscape scenes, and 8.4 points in science fiction scenes. These results indicate the model’s adaptability to varied structural and stylistic scene categories. Compared to other models, AMMFRM consistently achieved higher rendering quality across all scene types.

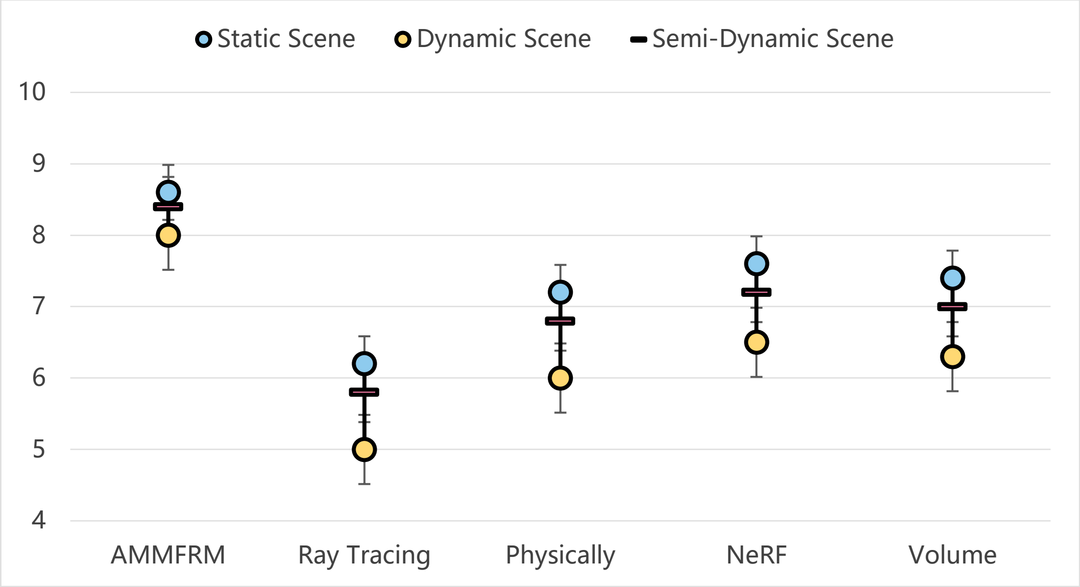

Figure 6: Rendering quality scores of scenes in different motion states (out of 10 points)

As shown in Figure 6, in the rendering quality test of scenes with different motion states, AMMFRM fully utilizes multimodal information in static scenes and scores 8.6 points. In dynamic scenes, although the difficulty increases, its semantic understanding and reasoning module can track the semantic changes of dynamic elements, and the rendering quality reaches 8.0 points. The semi-dynamic scene scores 8.4 points. The classic ray tracing model is difficult to calculate in dynamic scenes, and the score drops significantly, with 6.2 points for static scenes, only 5.0 points for dynamic scenes, and 5.8 points for semi-dynamic scenes. The physically based rendering model has difficulty adjusting the physical simulation of dynamic scenes, and the scores are 7.2 points, 6.0 points, and 6.8 points respectively. The NeRF model and the differentiable body rendering model are not as adaptable to dynamic scenes as AMMFRM, with scores ranging from 6.3 to 7.6 points.

Rendering quality scores are compared across static, dynamic, and semi-dynamic scenes to evaluate motion adaptability of the models.Figure 6 illustrates the semantic restoration results across the same three complex scene types. AMMFRM achieved 90% restoration in complex buildings, 93% in natural landscapes, and 91% in science fiction scenes, significantly outperforming the baselines in capturing accurate semantic relationships under diverse environmental structures.

Table 1: Semantic restoration degree of scenes in different motion states (%)

Model	**Static Scene**	**Dynamic Scenes**	**Semi-dynamic scene**
AMMFRM	93	88	91
Classic ray tracing model	72	60	68
Physically based rendering model	82	70	78
NeRF Model	86	75	82
Differentiable Volume Rendering Model	84	73	80

As shown in Table 1, in terms of semantic restoration of scenes in different motion states, AMMFRM has a semantic restoration of 93% in static scenes, 88% in dynamic scenes, and 91% in semi-dynamic scenes. The classic ray tracing model has insufficient understanding of the motion trajectory and interaction of elements in dynamic scenes, and has a low semantic restoration of 72% in static scenes, 60% in dynamic scenes, and 68% in semi-dynamic scenes. The physically based rendering model performs generally in dynamic scenes, with restorations of 82%, 70%, and 78% respectively. The semantic restoration of the NeRF model and the differentiable body rendering model in various motion state scenes is in the range of 73% - 86%, which is lower than AMMFRM.

Table 2: Rendering quality scores of scenes with different polygon counts (out of 10 points)

Model	**Low polygon scene**	**Medium polygon scene**	**High polygon scene**
AMMFRM	8.2	8.5	8.3
Classic ray tracing model	6.0	6.5	5.8

Model	Low polygon scene	Medium polygon scene	High polygon scene
Physically based rendering model	7.0	7.3	6.7
NeRF Model	7.5	7.8	7.6
Differentiable Volume Rendering Model	7.3	7.6	7.4

As shown in Table 2, in the rendering quality test of scenes with different polygon counts, AMMFRM maintained a high level in low, medium, and high polygon scenes. The low polygon scene scored 8.2 points, which can effectively use limited information for rendering; the medium polygon scene performed best, with a score of 8.5 points; although the high polygon scene was computationally complex, it still performed well, with a score of 8.3 points. The calculation amount of the classic ray tracing model increased dramatically in the high polygon scene, and the performance dropped significantly. The scores of low, medium, and high polygon scenes were 6.0 points, 6.5 points, and 5.8 points, respectively. The physically based rendering model, NeRF model, and differentiable rendering model are not as adaptable to different polygon scenes as AMMFRM, with scores ranging from 6.7 to 7.8 points.

Table 3: Scene semantic restoration degree of scenes with different numbers of polygons (%)

Model	Low polygon scene	Medium polygon scene	High polygon scene
AMMFRM	90	92	91
Classic ray tracing model	70	75	68
Physically based rendering model	80	83	77
NeRF Model	85	88	86
Differentiable Volume Rendering Model	83	86	84

As shown in Table 3, the semantic restoration test results of scenes with different polygon counts show that AMMFRM has a semantic restoration of 90% in low polygon scenes, 92% in medium polygon scenes, and 91% in high polygon scenes. The classic ray tracing model has a low semantic restoration of 70%, 75%, and 68% in each polygon count scene[23]. The physically based rendering model performs well in medium polygon scenes, with a restoration of 83%, and 80% and 77% in low and high polygon scenes, respectively. The NeRF model and the differentiable rendering model have a restoration of 83% to 88% in each polygon count scene, but are still inferior to AMMFRM.

Table 4: Rendering quality scores of different environment scenes (out of 10 points)

Model	**Interior scene**	**Outdoor scenes**	**Mixed indoor and outdoor scenes**
AMMFRM	8.4	8.3	8.5
Classic ray tracing model	6.1	5.9	6.0
Physically based rendering model	7.1	6.9	7.0
NeRF Model	7.5	7.4	7.6
Differentiable Volume Rendering Model	7.3	7.2	7.4

As shown in Table 4, in the evaluation of rendering quality in different environmental scenes, AMMFRM can accurately grasp the spatial layout and material semantics in indoor scenes, with a score of 8.4 points; it performs well in outdoor scenes, with a score of 8.3 points; in indoor and outdoor mixed scenes, the advantages of multimodal fusion and semantic understanding are more obvious, with a score of 8.5 points. The classic ray tracing model and the physically based rendering model have poor adaptability when switching between different environmental scenes, with a score of 5.9-7.1 points. A

4.3 Discussion

The superior performance of AMMFRM stems from its modular design, particularly the integration of a graph-based semantic reasoning module and adaptive rendering strategy generation. These allow the model to align rendering logic with scene semantics, outperforming traditional and deep learning methods. However, AMMFRM still faces challenges in extreme scenes with high-frequency motion or unusual lighting, where semantic misinterpretation may occur. Additionally, its reliance on multi-module deep learning structures demands higher hardware resources, which may limit scalability. The observed variance in results across FilmScene and GameWorld datasets can be attributed to differences in scene complexity—AMMFRM adapts more effectively in stylistically diverse film scenes than in high-interactivity game scenes, where real-time constraints are tighter[24].

To evaluate the design choices of key hyperparameters in AMMFRM,we conducted an ablation study focusing on the Visual Feature Perception Module(VFPM)and Semantic Comprehension and Inference Module(SCIM).For VFPM,varying the initial convolution kernel size from 3×3 to 7×7 showed that a 5×5 configuration offered the best balance between detail capture and computational efficiency.In SCIM,increasing GNN depth beyond 3 layers yielded diminishing returns and introduced noise in semantic propagation.Incorporating an attention mechanism improved semantic restoration by 4.2%,with multi-head attention(4 heads)performing optimally

5 Conclusion

This study focuses on solving the problems of existing 3D visualization scene rendering technology based on visual image language, and innovatively proposes AMMFRM. The research process covers the careful design of the model architecture and multi-dimensional experimental evaluation. AMMFRM has achieved remarkable results through testing on representative data sets. In terms of rendering quality score, it can reach 8.3 points for complex building scenes and 8.6 points for natural landscape scenes; the scene semantic restoration degree reaches 90% in complex building scenes and 93% in natural landscape scenes. Compared with classic ray tracing models, physically based rendering models, NeRF models, differentiable body rendering models, etc., AMMFRM has outstanding advantages. Its multimodal fusion mechanism and semantic understanding ability are the key to excellent performance. From the perspective of external validity and generalizability, the model performs stably in various scene types and has wide application potential. However, the experiment also reveals the limitations of the model in extreme scene adaptability and computing resource consumption.This study advances the theory of 3D visualization scene rendering and introduces new methods for related applications. In the future, the model can be further optimized by expanding the data set and exploring new hardware combinations to promote industry development.

Despite its strengths,AMMFRM has several limitations.The model's scalability is constrained by GPU memory when processing ultra-large scenes.Its performance may overfit to the structure of FilmScene and GameWorld datasets,requiring retraining for significantly different domains.Additionally,under highly dynamic conditions with unpredictable scene changes,semantic reasoning can lag,affecting real-time adaptability.

Fund

This work was sponsored in part by 2021 Shanxi Province Higher Education Teaching Reform and Innovation Project (J2021882)

References

[1] Kim S, Kang J. Voxel-wise UV parameterization and view-dependent texture synthesis for immersive rendering of truncated signed distance field scene model. Etri Journal. 2022;44(1):51-61. DOI: 10.4218/etrij.2021-0300

[2] Zhang YC, Gao Z, Sun WB, Lu Y, Zhu YH. MD-NeRF: Enhancing Large-Scale Scene Rendering and Synthesis with Hybrid Point Sampling and Adaptive Scene Decomposition. Ieee Geoscience and Remote Sensing Letters. 2024;21. DOI: 10.1109/lgrs.2024.3492208

[3] Shen CY, Chen CY, Hu XJ. Scene-content-sensitive real-time adaptive foveated rendering. Journal of the Society for Information Display. 2024;32(10):703-15. DOI: 10.1002/jsid.1346

[4] Stampfl V, Ahtik J. Quality of Color Rendering in Photographic Scenes Illuminated by Light Sources with Light-Shaping Attachments. Applied Sciences-Basel. 2024;14(5). DOI: 10.3390/app14051814

[5] Ni TL, Chen YS, Liu SP, Wu JL. Detection of real-time augmented reality scene light sources and construction of photorealis tic rendering framework. Journal of Real-Time Image Processing. 2021;18(2):271-81. DOI: 10.1007/s11554-020-01022-6

[6] Dai YR, Li J, Zhang Y, Jiang YQ, Qin HD, Zhou XS, et al. Scene-Constrained Neural Radiance Fields for High-Quality Sports Scene Rendering Based on Visual Sensor Network. Ieee Sensors Journal. 2024;24(21):35900-13. DOI: 10.1109/jsen.2024.3452436

[7] Mao D, Rao HY, Chen ZG, Wang JQ, Zhao S, Wang YD. Super-Resolution Virtual Scene Rendering Technology Based on Generalized Huber-MRF Image Modeling. International Journal of Computational Intelligence Systems. 2024;17(1). DOI: 10.1007/s44196-024-00619-0

[8] Ren HC, Huo YC, Peng YF, Sheng HT, Xue WD, Huang HX, et al. LightFormer: Light-Oriented Global Neural Rendering in Dynamic Scene. Acm Transactions on Graphics. 2024;43(4). DOI: 10.1145/3658229

[9] Sorin C, Mircea C, Diana C, Cristian G. A MATHEMATICAL MODEL AND AN EXPERIMENTAL SETUP FOR THE RENDERING OF THE SKY SCENE IN A FOGGY DAY. Revue Roumaine Des Sciences Techniques-Serie Electrotechnique Et Energetique. 2020;65(3-4):265-70.

[10] Xu JM, Wu XC, Zhu ZH, Huang QX, Yang Y, Bao HJ, et al. Scalable Image-based Indoor Scene Rendering with Reflections. Acm Transactions on Graphics. 2021;40(4). DOI: 10.1145/3450626.3459849

[11] Liu SH, Li MH, Zhang XN, Liu S, Li ZX, Liu J, et al. Image-Based Rendering for Large-Scale Outdoor Scenes With Fusion of Monocular and Multi-View Stereo Depth. Ieee Access. 2020;8:117551-65. DOI: 10.1109/access.2020.3004431

[12] McCormack L, Politis A, McKenzie T, Hold C, Pulkki V. Object-Based Six-Degrees-of-Freedom Rendering of Sound Scenes Captured with Multiple Ambisonic Receivers. Journal of the Audio Engineering Society. 2022;70(5):355-72. DOI: 10.17743/jaes.2022.0010

[13] Sun ZQ, Zheng HB, Lv CF, Bao JS. A fast scene geometric modeling approach for digital twins combining neural rendering and model retrieval. International Journal of Computer Integrated Manufacturing. 2025;38(4):501-19. DOI: 10.1080/0951192x.2024.2350539

[14] Kim S, Do J, Kang J, Kim HY. Rate-Rendering Distortion Optimized Preprocessing for Texture Map Compression of 3D Reconstructed Scenes. Ieee Transactions on Circuits and Systems for Video Technology. 2024;34(5):3138-55. DOI: 10.1109/tcsvt.2023.3310522

[15] Qin ZB, Chen Q, Qian K, Zheng QH, Shi JS, Tai YH. Enhancing endoscopic scene reconstruction with color-aware inverse rendering through neural SDF and radiance fields. Biomedical Optics Express. 2024;15(6):3914-31. DOI: 10.1364/boe.521612

[16] Park J, Cho K. Neural Rendering-Based 3D Scene Style Transfer Method via Semantic Understanding Using a Single Style Image. Mathematics. 2023;11(14). DOI: 10.3390/math11143243

[17] Zhu CJ, Zhang H, Liu QM, Zhuang ZX, Yu L. A Signal-Processing Framework for Occlusion of 3D Scene to Improve the Rendering Quality of Views. Ieee Transactions on Image Processing. 2020;29:8944-59. DOI: 10.1109/tip.2020.3020650

[18] Fan RZ, Shi XH, Wang KY, Ma QX, Wang LL. Scene-aware Foveated Rendering. Ieee Transactions on Visualization and Computer Graphics.

2024;30(11):7097-106. DOI: 10.1109/tvcg.2024.3456157

[19] Li ZP, Zhu JK. Point-Based Neural Scene Rendering for Street Views. Ieee Transactions on Intelligent Vehicles. 2024;9(1):2740-52. DOI: 10.1109/tiv.2023.3304347

[20] Wu XC, Xu JM, Zhu ZH, Bao HJ, Huang QX, Tompkin J, et al. Scalable Neural Indoor Scene Rendering. Acm Transactions on Graphics. 2022;41(4). DOI: 10.1145/3528223.3530153

[21] Dang P, Zhu J, Wu JL, Li WL, You JG, Fu L, et al. A real 3D scene rendering optimization method based on region of interest and viewing frustum prediction in virtual reality. International Journal of Digital Earth. 2022;15(1):1081-100. DOI: 10.1080/17538947.2022.2080878

[22] Rong MQ, Cui HA, Shen SH. Efficient 3D Scene Semantic Segmentation via Active Learning on Rendered 2D Images. Ieee Transactions on Image Processing. 2023;32:3521-35. DOI: 10.1109/tip.2023.3286708

[23] lmola SAS, Qasim NH, Alasadi HAA. Robust method for embedding an image inside cover image based on least significant bit steganography. Informatica. 2022;46(9):[Online-only issue]. doi:10.31449/inf.v46i9.4362.

[24] Li B, Sharma A. Application of interactive genetic algorithm in landscape planning and design. Informatica. 2022;46(3):[page range if available]. doi:10.31449/inf.v46i3.4049.