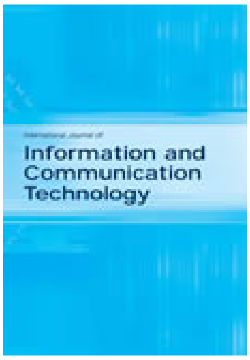


International Journal of Information and Communication Technology

ISSN online: 1741-8070 - ISSN print: 1466-6642
https://www.inderscience.com/ijict

Deep learning-driven real-time rendering technology for film and television animation special effects

Yuqi Tian, Fangqi Yang

DOI: 10.1504/IJICT.2025.10073072

Article History:

Received:	28 June 2025
Last revised:	23 July 2025
Accepted:	23 July 2025
Published online:	17 September 2025

Deep learning-driven real-time rendering technology for film and television animation special effects

Yuqi Tian*

School of Medias,
Weinan Normal University,
Weinan 714000, China
Email: crz220730@163.com
*Corresponding author

Fangqi Yang

School of Computer Science,
Weinan Normal University,
Weinan 714000, China
Email: yangfangqi@wnu.edu.cn

Abstract: Real-time rendering of film and television animation special effects meets the necessity to increase rendering efficiency and image quality with the advent of deep learning (DL) technologies. When handling complicated dynamic effects, conventional rendering techniques still struggle with long rendering times and unsteady image quality. This work proposes a DL-based optimisation technique to enhance the efficiency and image quality of dynamic effects generating using DL model. Compared with the conventional physics simulation rendering and particle system rendering, the experimental results show that the DL optimised rendering method not only greatly lowers the rendering time but also improves the image quality, especially in terms of detail performance and texture authenticity. This article presents future optimisation directions and offers a fast and high-quality solution for real-time rendering of film and television animation special effects.

Keywords: DL; film and television animation; special effects rendering; real-time rendering.

Reference to this paper should be made as follows: Tian, Y. and Yang, F. (2025) 'Deep learning-driven real-time rendering technology for film and television animation special effects', *Int. J. Information and Communication Technology*, Vol. 26, No. 33, pp.57–75.

Biographical notes: Yuqi Tian received her Bachelor's degree from the Lu Dong University in 2016, and Master's degree from the Chongqing University in 2019. Currently, she works in the Weinan Normal University. Her research interests are included film production, television production and communication.

Fangqi Yang received his Masters degree from the Northwest Normal University in 2011, and PhD degree from the Northwest Normal University in 2018. Currently, he works in the Weinan Normal University. His research interests include computer and educational technology.

1 Introduction

1.1 Background of study

Visual effects in the film and television business have reached hitherto unheard-of heights as worldwide technology and technological innovations in the domains of computer graphics, artificial intelligence (AI), and virtual reality (VR) fast advance. Animation effects have evolved from the first hand-drawn animation to today's computer-generated imagery (CGI) to become increasingly realistic and detailed, offering amazing visual impacts (Lamotte, 2022). Film and television production is confronting tremendous difficulties with efficiency and speed while striving for the best picture quality as the audience's desire for visual experience keeps growing. Particularly in the domains of interactive entertainment, VR and games, real-time rendering has evolved into a significant business bottleneck.

Usually, sophisticated physical simulation and computationally demanding rendering methods define the manufacturing process of cinema and animation. Though they may use a lot of computer resources and time, traditional rendering techniques such ray tracing and lighting models can provide quite realistic visual effects. Particularly for intricate dynamic scenes and highly detailed animations, conventional rendering techniques sometimes fall short in meeting the low latency and high frame rate required for real-time rendering, which is especially relevant in the fields of VR, real-time gaming, and film and TV effects production. Therefore, how to guarantee the quality of the picture while greatly increasing the rendering efficiency has become an urgent concern of the present film and television production technology.

Within this framework, DL, one of the fundamental technologies of AI, is transforming film and television animation into special effects creation by virtue of its strong data processing and pattern recognition features. Deep learning (DL) has considerably promised in picture rendering and special effects production since it can quickly create and optimise images through neural networks (Lee et al., 2021). Training neural networks enables DL to effectively render according to learnt characteristics, capture crucial features, texture information, and illumination changes in a scene, thereby greatly improving rendering speed while preserving image quality.

Especially in the acceleration of real-time rendering technologies, the use of DL in cinema and television animation special effects has been progressively deepened in recent years. Researchers have put forth several novel rendering techniques based on DL models like generative adversarial network (GAN) and convolutional neural network (CNN). These techniques generate near-realistic animation effects in a far quicker time and help to simplify physical simulation by teaching the models to produce high-quality images straight during the actual rendering process. DL-driven rendering approaches have major benefits over conventional rendering techniques that not only increase the rendering efficiency but also help to lower the hardware resource usage (Pandey et al., 2024).

Though DL has enormous promise for use in real-time rendering of film and television animation special effects, several issues still need to be resolved. First, the training of DL models depends on a lot of labelled data; however, the current film and TV animation datasets are sometimes inadequate to satisfy the demand for high-quality training. Second, the computational demand of deep neural networks is great, especially in high-resolution and high-detail scenes, and how to effectively leverage hardware resources to lower latency while assuring rendering effects is a challenging issue in the present technical development. Thus, a fundamental question in future research and practical applications is how to provide full play to the potential of DL in real-time rendering of film and television animation special effects and remove the technical hurdles it faces.

This work intends to investigate the real-time rendering technology of film and television animation special effects based on DL, so addressing the present performance bottlenecks and so improving the rendering quality so offering more effective and creative technical solutions for the field of film and television animation production.

1.2 Significance of study

This work addresses DL-driven real-time rendering technologies for cinema and television animation special effects, with great academic merit and practical application relevance. The scientific relevance of this study can be particularly summed up as follows:

1 Enhancing the rendering efficiency of film and television animation special effects: this work can considerably increase rendering efficiency and reduce the rendering time by adding the DL model. Particularly in application situations like VR and interactive film and television, this is of great relevance for real-time film and television animation production that calls for high frame rate and low latency.

2 Improvement of rendering quality and visual effects: DL-driven rendering techniques enable the accelerating rendering process without compromising image quality. This means that while guaranteeing great details and realistic light and shadow, film and television works can produce more efficient and smooth rendering effects and improve the audience's visual experience (Wang, 2025). For the technological innovation and quality enhancement of the film and television sector, this is of enormous relevance.

3 Promote the application of real-time rendering technology in interactive entertainment and VR: this work offers fresh technical concepts and approaches for the application of DL in various domains so that film and television animation special effects can not only be extensively used in conventional films and TV dramas but also be expanded to developing fields such VR, so supporting the diversified development of film and television production technology.

4 Provide technical basis for future intelligent film and television production: future film and television production will veer toward more intelligent and automated as AI technology develops constantly. By means of exploration and application of this research, it can offer theoretical basis and technical support for intelligent special effects generation, real-time rendering and other technologies in future film and television production and build the basis for intelligent industry development.

By means of these ramifications, this work not only advances the development of real-time rendering technology for film and television animation special effects but also offers fresh technical ideas and opportunities for future intelligent production of the film and television sector.

2 Deep learning

Important subfield of machine learning (ML), DL uses multi-level neural networks for data processing and feature learning by imitating the functioning mechanism of neurons in the brain. DL technology has great capacity in the processing of complicated data like images, audio, and text since it can learn the features of the data automatically without the necessity of manually constructing the features and so demonstrates a key advantage. Especially in the domains of computer vision, speech recognition, and natural language processing, DL technique has been extensively applied with the increase of computing capacity and the popularity of big data. Notwithstanding its outstanding performance in many disciplines, DL still has certain difficulties including high computing resource requirements, training data reliance, and model interpretability problems.

One of the most traditional models in DL, CNN, has performed exceptionally in image processing. CNN models the structure of the biological visual system to extract low-level elements in the image, such as edges and texture, by means of a convolutional layer; meanwhile, a pooling layer is utilised to lower the dimensionality of the input and therefore the computing work. CNN may progressively extract more sophisticated information, such object shape and structure, as the number of network layers rises (Liu et al., 2021). CNNs now accomplish notable performance in applications including image segmentation, target recognition, and image classification. CNN is not effective for temporal data or tasks with time dependency, (e.g., video analysis), even if it shines in processing still images. It also requires great preparation of the input image. Furthermore, one of the primary difficulties in applying the CNN model is the need for a lot of labelled data and computer resources required in its training.

Conversely, recurrent neural networks (RNN) are DL models intended especially to handle sequential data. RNNs convey prior states from each moment to the present moment, therefore capturing temporal dependencies in sequences unlike those of conventional neural networks. This lets RNNs shine in chores such natural language processing, speech recognition, and time-series data processing. RNNs have mostly benefited from their capacity to effectively interpret input sequences of arbitrary length, which has resulted in important uses in fields such machine translation and speech-to-text (Kumar et al., 2023). RNN does, however, also have certain issues, particularly regarding long time dependencies; it is prone to the phenomena of gradient vanishing or gradient explosion, which causes training difficulties.

Long short-term memory (LSTMs) networks have surfaced as a solution for this challenge. Dealing with extended temporal dependencies, LSTMs, a development of RNNs, overcome the limitations of the conventional RNNs. By means of input gates, forgetting gates, and output gates, which govern the storage and forgetting process, LSTMs manage the flow of information, therefore enabling the network to preserve memory over an extended period. LSTM's ability to perform sentiment analysis, text production, and speech recognition makes it better in all around activities (Li et al., 2020). Although LSTM provides great benefits in modelling extended sequences, its training method is still complicated and has a substantial computational overhead. Furthermore, LSTM could not be as successful as intended in some situations since memory degradation could still occur under very long sequences.

Conversely, GAN is a DL model created by adversarial training that creates data. A generator and a discriminator make up GAN. While the discriminator aims to separate produced data from actual data, the generator seeks to produce data as realistic as feasible. This game motivates the generator to constantly improve the quality of the produced data, thereby producing quite realistic data at last. In terms of data augmentation, style migration, and image generation, GAN has produced notable success. Its primary benefit comes from its capacity to create quite lifelike data and visuals. To get the intended outcomes, GAN's training process is somewhat erratic and requires a lot of training data, nevertheless. Furthermore, lacking interpretability and black-box models, GAN models can present a challenge in several high-risk fields like banking and healthcare.

Often used for data reduction and feature learning in unsupervised learning, autoencoder (AE) compresses the input data into a low-dimensional latent space representation and subsequently reconstructs the original data from that representation, hence achieving denoising or data compression. AE is extensively used for data noise reduction, anomaly detection and picture denoising and other applications; it may be trained without annotated data. An expansion of AE, variational autoencoder (VAE) introduces a probabilistic model that makes the produced results more varied and continuous, hence improving performance in generative tasks (Huang et al., 2021). While AE and VAE shine in unsupervised learning, their generative powers are restricted, particularly in relation to complicated, high-dimensional data and the quality of the produced samples usually suffers relative to that of GANs.

Mostly applied for decision-making problems, reinforcement learning (RL) is a significant subfield in DL. RL is essentially about learning a strategy by interacting with the surroundings. Maximising the total reward will help one to choose the best action. For jobs requiring constant decision-making, such game intelligence, autonomous driving and robot control, RL is especially appropriate. Deep reinforcement learning (DRL) has proved successful in many challenging tasks and combines the benefits of DL and RL in recent years to make efficient decisions in high-dimensional state spaces (Pattanayak et al., 2024). Although DRL can handle increasingly difficult decision-making issues, its training approach is typically rather time-consuming and computationally costly. Furthermore, challenges in present work are DRL model stability and convergence.

With diverse models having their own strengths and constraints, DL models overall have shown great ability in applications in many spheres. DL still has issues including high data dependency, high computing resource consumption, and poor model interpretability even if it has produced innovations in numerous activities. Deeper study and technical advancement will help DL be more important in more different fields.

3 Real-time rendering techniques for animation effects

Using CGI technology, surreal, fantasy or natural visual effects in films, TV shows, video games, etc. are created from film and television animation special effects. These effects substantially improve the visual expression and audience immersion: smoke, fire, liquid simulation, explosions, light and shadow effects, dynamic changes of actors and objects, etc. The development of cinema and television animation special effects involve sophisticated 3D animation, physical simulation and real-time interaction rather than restricted simple static visuals as technology advances constantly. Modern film and television production has progressively high standards for special effects, needing visuals to keep detail and realism while assuring the speed of rendering to respond to the needs of real-time interaction and fast updating.

In a virtual environment, real-time rendering technology is the ability of rendering techniques to create visuals with minimal or no latency and updates them in real-time. Unlike conventional offline rendering techniques, real-time rendering needs a frame to be computed in a very short amount of time, typically requiring the rendering time to be managed at the millisecond level, therefore adjusting to application situations including games, VR and interactive cinema and television. To guarantee good image quality and preserve the smoothness of the image, the system must rapidly handle a lot of data and calculation in every frame in real-time rendering. Real-time rendering methods may depend on strong graphics processing unit (GPU) to speed the rendering process via parallel computing to reach this aim (Jiménez de Parga and Gómez Palomo, 2018).

Although conventional rendering techniques including ray tracing and path tracking can produce quite realistic light and shadow effects and material details in film and television animation special effects production, their computational volume is so great that it is challenging to meet real-time needs. Many researchers and engineers have suggested several optimisation strategies to speed up the rendering process and raise performance to meet this difficulty. For instance, a rendering technique combining hardware acceleration with rasterisation methods keeps enough image quality while raising speed (Kivi et al., 2022). By pre-computing and removing invisible components, rasterisation, the process of turning a 3D visual into a 2D image, can greatly cut the amount of work required. Because of its efficiency, rasterisation, which reduces image quality compared to ray tracing, has become among the most often utilised methods in real-time rendering.

Furthermore, the suggestion to reduce reliance on processing resources while rendering are certain acceleration algorithms and approximation strategies. For instance, radiance estimate, and optimisation of rapid lighting models enable real-time rendering to apply complicated lighting computations (Liu et al., 2022). The advent of methods like light mapping and ambient occlusion (AO) not only dramatically speeds the rendering speed but also efficiently improves the light and shadow expression in the scene, so boosting the realism of the picture. These methods increase the general rendering performance by pre-computing and caching some image characteristics, therefore lowering the number of components that must be recalculated during rendering.

Dynamic scenery and sophisticated physics models provide more difficulties for real-time rendering. Their real-time rendering is especially challenging since special effects simulations such smoke, flames, liquids, and soft-body physics frequently involve a lot of particle systems and sophisticated physics computations. Researchers have created a range of particle system acceleration techniques, including particle swarm optimisation (PSO) and smoothed particle hydrodynamic (SPH)-based methods, which allow these effects to be adapted to real-time rendering requirements while maintaining their efficacy by simplifying complex physics simulations into discrete particles for real-time processing, so overcoming these challenges.

Apart from conventional techniques of rendering, new hardware technologies have greatly improved real-time rendering. GPU-based parallel computing, for instance, greatly speeds rendering efficiency by allowing the rendering process to manage several picture pixels and computational chores simultaneously (Elshakhs et al., 2024). Concurrent with this has been a significant advance in recent years: real-time ray tracing technology. While ray tracing is more computationally demanding, using hardware acceleration, e.g., NVIDIA's RTX graphics cards, and optimisation techniques allows real-time ray tracing to fit real-time rendering demands while preserving realistic lighting and shadow effects. Modern games and film and television production have extensively embraced this hardware-supported real-time ray tracing technology, so improving visual experience.

Though real-time rendering technology has advanced greatly, large-scale scenes and highly complex animations present several difficulties. Still one of the challenges is how to guarantee rendering efficiency while drawing in high detail, particularly on how to lower the hardware overhead. Researchers are striving to solve this issue by creating more effective image optimisation and acceleration techniques including intelligent compute allocation strategies in dynamic scenes and multi-level rendering approaches. Furthermore, the effects of real-time rendering, that of lighting, shadows, reflections, etc. still has to be improved to improve visual realism and immersion.

All things considered, the real-time rendering technology for film and television animation special effects is always changing toward more efficiency and realism as hardware technology and new algorithms arise. Future real-time rendering will more precisely mix the real and virtual, thereby offering a smoother and more immersive experience for film and television production, VR, and other interactive applications, even if obstacles still exist with the ongoing development of technology.

4 DL-driven real-time rendering of film and television animation special effects

Film and television animation special effects use real-time rendering technology, which comprises of multiple modules each assigned to a particular rendering choreography. Among these modules are scene modelling, texture mapping, lighting computation, physics simulation, special effects creation, and final image compositing. To get effective and seamless dynamic effects, each module must satisfy the criteria of real-time rendering and guarantee the rendering quality see Figure 1.

Film and television animation special effects mostly rely on the following basic modules for real-time rendering technologies.

4.1 Scene modelling and geometry construction

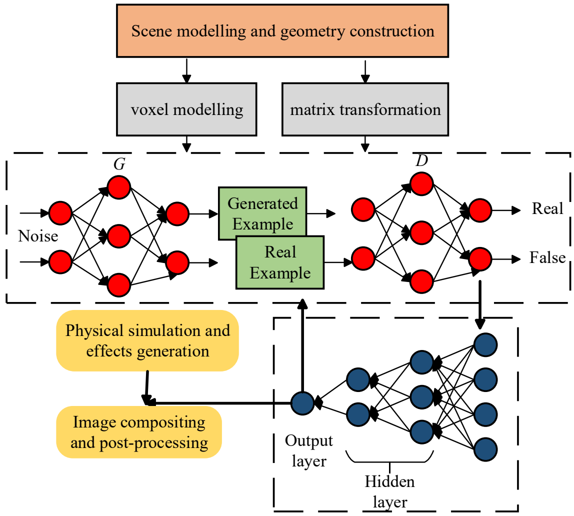
In this process, modelling methods based on voxel representation are utilised. Small cubes in 3D space, voxels are spatially meshed to convert geometric form of an item into volumetric data. While ensuring better accuracy, voxel-based modelling is more flexible in handling constantly changing and complicated shaped objects than conventional polygonal mesh representation, therefore substantially simplifying the modelling process.

A voxel representation of an object provides the position and state of each voxel using spatial coordinates (x, y, z) and flag bits (e.g., 0 or 1). The geometry of each item is represented by a huge collection of voxels V. The coordinates of each voxel in the voxel set can be stated by the following equation:

$$V = \{(x_i, y_i, z_i) | 1 \leq i \leq n\} \tag{1}$$

where V is the set of voxels; (x_i, y_i, z_i) are the spatial coordinates of the i^{th} voxel; n is the overall count of voxels. Coordinate transformations let it effectively move and update objects using this voxelised view.

Figure 1 Real-time rendering technology for film and television animation special effects (see online version for colours)



Matrix operations allow one to accomplish transformations of an object throughout the modelling process. When translating, rotating, or scaling an object assuming P as its voxel coordinates, the changed coordinates are obtained by the transformation matrix M updating each voxel coordinates:

$$P = (x, y. z, l)^T \tag{2}$$

$$P' = M \cdot P \tag{3}$$

where M is the transformation matrix; P' are the changed voxel coordinates. In this sense, one may precisely regulate the object's spatial position and attitude.

In applications, the effectiveness of voxel modelling is directly linked to the level-of-detail (LOD) optimisation technique, which dynamically changes the resolution of objects at different distances therefore reducing the amount of pointless computation during rendering (Chen et al., 2018). Low-resolution voxel mesh can be used to depict objects far away, vice versa with a high-resolution mesh. The rendering efficiency can be raised while guaranteeing the rendering quality by separating the voxel mesh and dynamically changing the voxel density in several areas.

All things considered, the voxel-based scene modelling technique allows straightforward, effective voxelised representations to enable the fast building and display of intricate 3D objects and scenes. Together with voxel transformation methods and detail level optimisation, it can significantly increase the efficiency of real-time rendering and give strong geometric data support for later rendering and physical simulation.

4.2 Texture mapping and material generation

This is a fundamental stage in the real-time rendering of film and television animation special effects and overseas adding details to a 3D object, so it seems realistic. By precisely affining a 2D picture (texture) to the surface of a 3D object, texture mapping methods replicate the intricacies and characteristics of an object's surface. Conversely, material generation is the definition of the optical characteristics of an item's surface, including reflection, refraction, transparency, and roughness, which directly influences how the object interacts with the light source, so influencing the rendering.

In texture mapping, every surface point of a three-dimensional object is mapped to a two-dimensional texture image's coordinate. This procedure uses a texture coordinate system, whereby the mapping rules map each vertex on the object surface to a pixel in the texture image. The mapping relation shows the equivalent 2D texture coordinates (u, v) assuming a vertex of a 3D object has coordinates (x, y, z):

$$(u, v) = f(x, y, z) \tag{4}$$

where f x, y, z) is the 3D space point mapping mechanism translating points to the 2D coordinates of the texture image. This method lets 3D objects have sophisticated surface textures without introducing extra geometric complexity.

The physically based rendering (PBR) system is applied in the rendering process to replicate the interaction of the surface of an object with light for material generating (Niu and Lo, 2022). Mainly metallicity, roughness, and reflectivity, the PBR material system defines the optical characteristics of an object's surface using a set of physical criteria. The following equation helps one to understand how an item surface reflects light, particularly with reference to low light angles and reflection enhancement:

$$F = F_0 + (1 - F_0)(1 - \cos\theta)^5 \tag{5}$$

where F is the reflection coefficient; F_0 is the reflection coefficient in the direction normal to the surface of the object; θ is the angle formed between the incident light and the normal surface. Particularly at smaller incident angles, this formula reflects the physical law of reflection phenomenon, hence, greatly increases the object surface reflectivity.

GAN is applied in texture generation to create high-quality texture images by training a generator and a discriminator network so that the generator may produce realistic texture samples from noise and the discriminator is in charge of ascertaining whether the produced texture is realistic enough. One may represent the GAN training process using the following formula:

$$L_{GAN} = \mathbb{E}_{x \sim p_{data}(x)}[\log D(x)] + \mathbb{E}_{z \sim p_z(z)}[\log(1 - D(G(z)))] \tag{6}$$

where x denotes the real data; z denotes the random noise input to the generator; $D(x)$ is the discriminator's decision on the real data; $D(G(z))$ is the texture image output from the generator; $G(z)$ is the random noise input to the generator. Through optimal generator and discriminator performance, this loss function helps the generator create ever realistic texture images.

Multiple texture layers together help to improve rendering in difficult scenarios. These comprise the base colour texture, the normal map and the specular map, which respectively regulate the colour, surface bump and gloss of the item. The performance of several materials may be exactly realised by adjusting these texture layers; the DL approach improves the diversity and details of texture production, so increasing the realistic material performance.

In real-time rendering of film and television animation special effects, texture mapping and material production ultimately form the main links. By means of accurate texture mapping, PBR material system, and GAN generation methods, 3D objects may be provided with rich details and highly realistic optical qualities, so improving the visual expression of the virtual world (Kuang et al., 2023).

4.3 Light calculation and environment modelling

This module uses CNNs to automatically learn and optimise the lighting and environmental effects in the scene, therefore enhancing the rendering efficiency and quality. Particularly, the CNN can automatically estimate and compute the light intensity depending on the input scene data including ambient circumstances, object surface attributes, and light source position. Without explicitly computing the path of every light ray, CNNs trained to detect various lighting conditions and object surface attributes may predict intricate effects including reflection and refraction of light in real-time during rendering. This not only increases computing efficiency but also helps to prevent some of the presumptions mistakes in conventional lighting designs.

CNN typically uses several convolutional layers to extract local properties of light and delivers the final light intensity and surface reflection information through fully connected layers when handling low computation (Zuo et al., 2022). Reducing the error between the projected and actual light will help to maximise its training process, the loss function can be stated using the following equation:

$$L_{lighting} = \sum_{i=1}^{n} \left(\hat{I}_i - I_i \right)^2 \tag{7}$$

where I_i is the real light intensity; $\hat{I}_i$ is the light intensity predicted by the network; n is the number of training samples. Through optimal loss function optimisation, the network can faithfully forecast the reflection and refraction effects under various illumination environments.

CNN is applied in terms of environmental modelling to handle AO consequences in the scene. By automatically creating optimal ambient light effects, considering the distribution of the light sources in the scene, the occlusions and reflections between the objects, and other elements, the CNN network can handle complicated ambient light occurrence patterns. The training goal is to reduce the error between the true value and the projected ambient light occlusion value; the loss function may thus be stated as:

$$L_{AO} = \sum_{i=1}^{n} \left(\hat{A}O_i - AO_i \right)^2 \tag{8}$$

where $\hat{A}O_i$ is the expected ambient light occlusion value, AO_i is the actual value, and n is the total training sample count. By means of CNN network optimisation, the accuracy of ambient light masking may be raised, so strengthening the realism of the scene (Marques et al., 2022).

Especially in dynamic scenes, CNN can significantly increase the efficiency and accuracy of the rendering effect by means of these techniques; it can also adjust and optimise the lighting and shadow effects in real-time, so avoiding the constraints of conventional manual design and offering more flexible and accurate rendering capabilities, so enabling revolutionary progress to the real-time rendering technology of film and television animation special effects.

4.4 Physical simulation and effects generation

Often including the creation of dynamic effects such fluids, smoke, flames and particles, this is a fundamental feature of cinema and television animation. GANs are utilised to speed this process and create more realistic and efficient special effects since conventional physics simulation techniques are computationally demanding and challenging to depict in real-time.

Training a GAN lets the network create fluid dynamics straight from the input scene data, hence, lowering the need to calculate particles one by one. The generator of the GAN creates a fluid simulation depending on the illumination, object form, and other scene circumstances. Unlike conventional approaches, GAN can generate the effects by automatically learning the fluid's movement patterns in many contexts without the need of accurately computing the state of every physical particle (Brunton et al., 2020). The network is trained with the aim of decreasing the discrepancy between the produced effect and the real fluid simulation, therefore optimising the fluid generating process. Formulated as follows is the fluid simulation's optimisation goal:

$$L_{fluid} = \left\| \hat{F} - F_{real} \right\|_2^2 + \lambda \cdot \mathbb{E}[\log(1 - D(G(z)))] \tag{9}$$

where F_{real} is the actual fluid effect obtained by conventional physical simulation; $\hat{F}$ is the fluid effect predicted by the network; $D(G(z))$ is the outcome of the discriminator's evaluation of the produced fluid; λ is a hyperparameter balancing the generating and discrimination losses. The network may learn and create more realistic fluid effects with this optimisation formulating.

By using these approaches, GAN is possible to create high-quality dynamic effects lowering the computational complexity in conventional physics simulations. Apart from accelerating the computing process, GAN enhances the realism and expressiveness of the produced effects in comparison with conventional simulation techniques.

4.5 Image compositing and post-processing

Involving responsibilities including combining several picture sources, colour correction, augmentation of light and shadow effects, and detail modification, this is a fundamental stage in film and animation production. To guarantee a natural and seamless effect, this technique not only calls for excellent image synthesis but also real-time data processing of a great volume.

CNNs are applied in image synthesis to automatically learn visual properties including texture and illumination, therefore producing a synthetic image. Extensive details and natural transitions are produced by the network from extracting features from the input image. The network's training goal is to reduce the generated image's difference from the target image.

CNNs are used in style migration to separate and reinterpret image content and style. The network first extracts the style information from the source image then applies it to the target image to create an image with artistic impact. Style migration has two components: content loss and style loss; so, by reducing the variation between these two components, the network can balance the conversion between content and style:

$$L_{style} = \|F_s - F_t\|_2^2 + \lambda \|C_s - C_t\|_2^2 \tag{10}$$

where C_s and C_t are content features; F_s and F_t are the stylistic elements of the source and target images; λ is a hyperparameter that modulates the weights of content and stylistic losses.

CNNs are used for colour correction and enhancement in post-processing to automatically adjust the hue and brightness of an image therefore enhancing its naturalness. Reducing the disparity between the produced image and the true image will help to define the loss function for colour enhancement:

$$L_{color} = \sum_{i=1}^{n} \|\hat{I}_i - I_i\|_2^2 \tag{11}$$

Furthermore, frequent post-processing chores where CNNs may automatically remove noise and enhance image resolution by learning noise patterns in the image include denoising and super-resolution (Sajjanar et al., 2024). CNN learns the mapping link between low-and high-resolution pictures in the super-resolution job, therefore producing a higher resolution image and recovering more features.

In summary, CNN applied in image synthesis and post-processing greatly enhances the quality and processing efficiency of images, therefore obtaining more detailed and lifelike results.

5 Experimental results and analyses

5.1 Experiments and analysis of results

Blender dataset is selected as the fundamental dataset in this work to assist the investigation of DL-driven real-time rendering technologies for cinema and television animation special effects.

Table 1 Information about the blender dataset

Dataset name	*Blender dataset*
Data types	3D scene models, textures, lighting data, rendering results, dynamic effects
Dataset size	Contains multiple 3D models and scenes with extensive lighting, material, texture, and rendering data
Key features	Provides complete 3D scene data, including environments, objects, textures, lighting, and animation effects, supporting various rendering tasks and DL model training.
Target tasks	Scene modelling, texture mapping, lighting calculation, physical simulation, animation generation, image synthesis and rendering

Blender dataset comprises environment models, object models, textures, lighting information and rendering results from several real-world and virtual environments. These facts make it quite relevant for the research on real-time rendering methods for film and television animation special effects. Perfect for DL training, the dataset spans a broad spectrum of elements from static scene modelling to dynamic rendering. Table 1 shows the blender dataset details.

5.2 Experimental environment

The experimental environment of this work is set with high-performance computing devices and adopts modern DL frameworks and development tools to guarantee the efficiency and scalability of the experiments, so enabling the research on DL-driven real-time rendering technology for film and television animation special effects.

Regarding hardware, the primary computer used in this research is a workstation with a 1TB NVMe SSD, 32 GB of DDR4 RAM, an NVIDIA RTX 3090 graphics card, the most recent Intel Core i9-11900K CPU. Particularly in complicated 3D models and rendering data, this hardware arrangement offers enough computing capability for DL training; the strong performance of the graphics card can greatly speed the training process. With 24 GB of graphics RAM to enable high-resolution image rendering and large-scale DL model training, the RTX 3090 graphics card offers.

From the software standpoint, this work makes use of the mainstream DL field frameworks, TensorFlow and PyTorch, which offer adaptable neural network model construction tools to enable the application of several DL algorithms. While PyTorch is mostly utilised for lighting calculation and special effects, TensorFlow is built and trained CNNs for scene modelling, texture mapping, and other modules. Model development in generation, whose dynamic computational graph advantage makes it more appropriate for processing dynamic picture data. Furthermore, employed in this work were the CUDA toolkit and the cuDNN library to maximise the computational efficiency of the DL model, particularly under GPU acceleration, thereby enabling faster processing of vast-scale datasets.

This work uses highly compatible and stable Ubuntu 20.04 LTS for the operating system, which also extensively supports DL frameworks and development tools. Furthermore, installed for data processing, image processing, and result visualisation were Python 3.8 environment settings with relevant DL libraries such as NumPy, SciPy, OpenCV, Matplotlib and so on (Pontén et al., 2024).

Furthermore, this work uses Docker container technology to guarantee the consistency and compatibility of the experimental environment thereby attaining more effective training and testing. The containerised deployment guarantees the stability and repeatability of the experimental results, lowers configuration variations between several systems, and helps management of the experimental environment and dependencies.

5.3 Experimental procedure

Two tests were developed in this work to validate the efficacy of DL-driven real-time rendering approaches for film and television animation special effects. While the second experiment concentrates on the application of DL in dynamic special effects creation and real-time rendering, the first experiment assesses DL-based scene modelling and rendering optimisation strategies.

The first experiment intends to confirm the efficiency of DL-based scene modelling and rendering optimisation methods. This work constructs a DL model based on CNN and GAN using stationery and dynamic scene data from blender dataset for scene modelling and texture mapping. By using training, the DL network can automatically detect items in the scene, create matching textures and materials, and improve rendering effects and illumination.

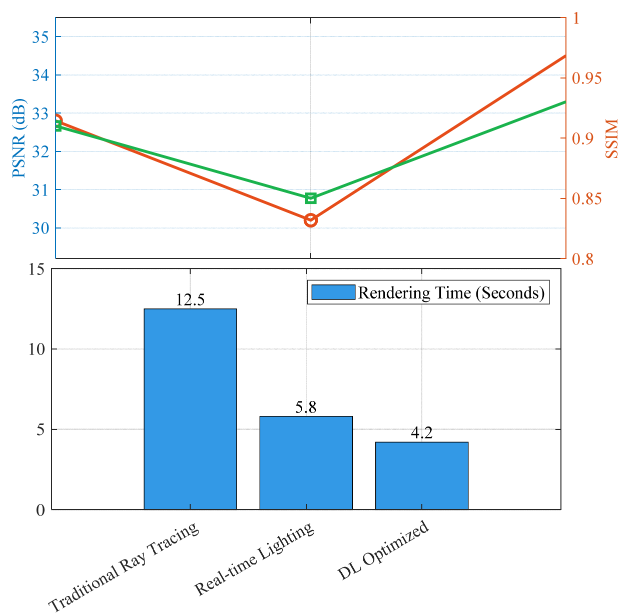
Images were first created using blender using conventional rendering techniques, (e.g., ray tracing and real-time light rendering) and subsequently the same scene was created using the trained DL model during the experiment. To guarantee the equity of the experiment, scene complexity, rendering resolution, and lighting conditions were regulated during the rendering process. The experiments documented the time needed for every technique to create the same scene by means of comparison between several rendering techniques, therefore quantitatively assessing the image quality in the performance evaluation session. Three metrics which consist of rendering time (in seconds), peak signal-to-noise ratio (PSNR, dB), and structural similarity index measure (SSIM), are utilised in the quantitative evaluation to evaluate the image rendering speed and quality (Al Najjar, 2024).

While SSIM mostly measures the structural similarity of an image, with higher values representing an image that is more similar to the original image in terms of visual perception; PSNR is used to measure the degree of noise and distortion in an image; higher values represent better image quality. In terms of detail retention, texture representation, and lighting simulation, the thorough analysis of these measures may fairly assess the benefits of the DL optimised rendering technique.

Figure 2 displays the experimental findings.

From the experimental results, it is clear that in all the assessed criteria the DL-optimised rendering approach beats the conventional rendering technique. First of all, in terms of rendering time, which dropped by around two-thirds from 12.5 seconds to 4.2 seconds, DL-optimised rendering much exceeds conventional ray tracing rendering. This result illustrates that, particularly in complicated scenarios, DL-optimised rendering can greatly increase rendering efficiency and drastically lower calculation time.

Figure 2 Comparison of different rendering methods (see online version for colours)



With PSNR rising from 32.8 dB in conventional ray-traced rendering to 34.5 dB, DL-optimised rendering shows good image quality in both PSNR and SSIM measurements, therefore preserving more detail and efficiently lowering image noise. The SSIM also shows improvement from 0.91 to 0.93, suggesting that the DL-optimised rendering is noticeably more structurally like the source image and that the visual impression is enhanced generally.

Though the rendering speed of the real-time light rendering technique is faster, its major drawback in image quality is evident. Its PSNR is 30.2 dB and SSIM is 0.85, therefore indicating a decline in image quality relative to DL-optimised rendering. While real-time rendering technologies can finish the rendering job rapidly, their clear benefits in terms of detail performance and lighting simulation outweigh their shortcomings. Therefore, DL-based rendering optimisation not only increases the rendering efficiency in terms of speed but also makes major development in terms of image quality.

In general, DL-optimised rendering techniques perform rather well in stationary scenes and basic lighting conditions; yet conventional rendering techniques still have benefits in more challenging dynamic scenarios and physical effects. The next study will concentrate on the application of DL in dynamic effects generation and real-time rendering, notably on the difficulties of physics simulation and real-time generation of complicated effects, thereby addressing this issue.

The objective of the second experiment is to validate in managing complicated effects and dynamic scenes the efficacy of the DL-driven dynamic effects generating and real-time rendering methodologies. The scenario is rendered and optimised in the experiment using dynamic scene data from blender dataset comprising smoke, flame, fluid and other physical effects.

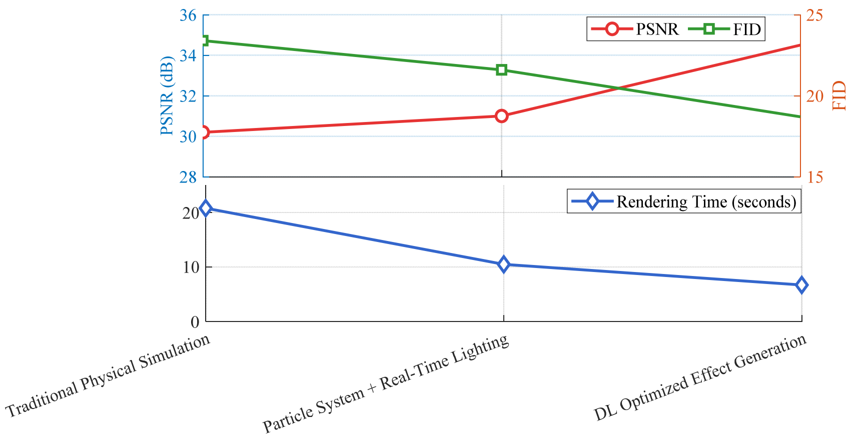
This experiment aims to investigate the efficacy of DL-optimised rendering techniques in producing complex physical effects, (e.g., fluid simulation, smoke, flames, etc.), and to compare them with conventional physical simulation methods (particle systems, real-time light rendering, etc.). Thus, rendering time and image quality will be evaluated.

First, the experiment chooses several blender dataset datasets with dynamic sceneries and effects encompassing a range of dynamic effects such smoke, flame, liquid flow, and so on. Then, by means of the training of the dynamic effects, the DL model learns and generates them, therefore enabling the model to produce more lifelike and detailed animation effects. After that, the generated dynamic effects using the trained DL model are compared with conventional physical simulation rendering techniques (like particle system + real-time light rendering). Indices including rendering time and image quality help to assess the performance of the DL-optimised rendering technique. Figure 3 displays the experimental outcomes.

The experimental findings clearly show that the method of DL-optimised special effects creation shines in image quality and rendering time. First, the 6.7 seconds of DL-optimised rendering are far less than the 20.8 seconds of conventional physics simulation rendering and the 10.5 seconds of particle system + real-time lighting rendering. This indicates that, fit for real-time rendering needs, the DL optimisation method considerably increases the rendering efficiency and may complete the development of complicated special effects in a shorter amount of time.

The PSNR of the DL-optimised special effects method is 34.5 dB, which is much higher than that of the conventional physics simulation rendering (30.2 dB) and the particle system + real-time lighting rendering (31.0 dB), thus the increase in the PSNR value means that the DL-optimised rendering generates a significantly better image quality than the other methods with sharper details, less noise, and a higher quality of the image. Furthermore, the FID value of 18.7 is much lower than the 23.4 of the conventional method and the 21.6 of the particle system + real-time lighting rendering, indicating that the DL-optimised special effects generation not only improves the quality, but also generates special effects with more degree of resemblance to the real scene and more natural effects.

Figure 3 Performance comparison of different rendering methods (see online version for colours)



With a PSNR of just 30.2 dB and an FID of 23.4, the conventional physics simulation rendering can produce some dynamic special effects, but its rendering time is longer, and the image quality is inferior, hence lacking detail and realism in the special effects. This could be because the optimisation of rendering is not as flexible as DL methods and conventional techniques are challenging to deal with details in complicated situations.

With a PSNR of 31.0 dB and an FID of 21.6, the quality of the image is still inferior even if the particle system + real-time lighting rendering method beats the conventional physics simulation rendering in terms of rendering time. This implies that, although the particle system and the real-time lighting rendering can speed up the rendering process, they are far inferior in terms of generating the details, textures and realism of the complex physics special effects. The DL technique generates more realistic and detailed dynamic scenes in the actual world.

All things considered, especially in the creation of complicated dynamic effects, DL-optimised effects generating techniques not only greatly lower rendering time but also greatly increase image quality and detail retention. For real-time rendering and development of premium film and television animation special effects, this offers a practical answer. Future studies can tune the DL model to increase its performance in more complicated scenarios, particularly in the formation of fluid, smoke and flame effects.

6 Conclusions

This work offers a dynamic special effects generating method based on DL optimisation, systematically investigates DL-driven real-time rendering technology for cinema and television animation special effects and compares it with conventional physical simulation rendering technology. The experimental findings reveal that in terms of rendering speed and image quality the DL-optimised rendering technique has notable benefits. Especially in the processing of complicated scenes and dynamic physical special effects, which shows great potential, the DL-optimised special effects generating model not only greatly increases rendering efficiency but also generates more realistic and detailed dynamic effects.

Although the DL optimisation approach offers great performance in terms of image quality and rendering efficiency, real applications nevertheless suffer significant restrictions. First, a lot of high-quality training data is needed for the training of DL models, so in some special effects fields it could become a bottleneck. Second, the computational cost of DL models is significant, particularly in the processing of intricate scenarios that call for a lot of computer resources, therefore restricting their use in some low-resource situations. Finally, even although DL can improve the quality of dynamic special effects, there are still some difficulties including the room for additional optimisation in terms of very high rendering accuracy and detailed representation of complicated physical special effects.

Future studies should probe this thoroughly in numerous angles. First, the usage of data augmentation approaches or unsupervised learning methods can be investigated to increase the generalisation ability of the model in several environments to solve the issue of inadequate training data. Second, improving the efficiency of useful applications depends much on optimising the computational efficiency of DL models and lowering their computational load. Technical methods such model compression, quantisation and hardware acceleration help to increase the computational speed and real-time rendering capability of the model (Deng et al., 2020). Moreover, as DL technology develops constantly, DL models coupled with physics engines will also become a hot topic in the future, investigating how to replicate physical special effects more precisely and accomplish more natural animation effects with DL.

Future studies should also concentrate on the merging of DL and conventional rendering techniques leveraging the benefits of both to offset their respective drawbacks. Especially under the demand of real-time rendering, combining DL with traditional rendering techniques offers a more flexible way to improve image quality while preserving efficient rendering. DL-driven real-time rendering technique for film and television animation special effects will become ever more significant in film and television production, gaming development and other sectors as processing power increases and model optimisation advances.

Declarations

All authors declare that they have no conflicts of interest.

References

Al Najjar, Y. (2024) 'Comparative analysis of image quality assessment metrics: MSE, PSNR, SSIM and FSIM', *International Journal of Science and Research (IJSR)*, Vol. 13, No. 3, pp.110–114.

Brunton, S.L., Noack, B.R. and Koumoutsakos, P. (2020) 'Machine learning for fluid mechanics', *Annual Review of Fluid Mechanics*, Vol. 52, No. 1, pp.477–508.

Chen, Q., Liu, G., Ma, X., Mariethoz, G., He, Z., Tian, Y. and Weng, Z. (2018) 'Local curvature entropy-based 3D terrain representation using a comprehensive Quadtree', *ISPRS Journal of Photogrammetry and Remote Sensing*, Vol. 139, pp.30–45.

Deng, L., Li, G., Han, S., Shi, L. and Xie, Y. (2020) 'Model compression and hardware acceleration for neural networks: a comprehensive survey', *Proceedings of the IEEE*, Vol. 108, No. 4, pp.485–532.

Elshakhs, Y.S., Deliparaschos, K.M., Charalambous, T., Oliva, G. and Zolotas, A. (2024) 'A comprehensive survey on Delaunay triangulation: applications, algorithms, and implementations over CPUs, GPUs, and FPGAs', *IEEE Access*, Vol. 12, pp.12562–12585.

Huang, Y., Tang, Y. and VanZwieten, J. (2021) 'Prognostics with variational autoencoder by generative adversarial learning', *IEEE Transactions on Industrial Electronics*, Vol. 69, No. 1, pp.856–867.

Jiménez de Parga, C. and Gómez Palomo, S.R. (2018) 'Efficient algorithms for real-time GPU volumetric cloud rendering with enhanced geometry', *Symmetry*, Vol. 10, No. 4, p.125.

Kivi, P.E., Mäkitalo, M.J., Žádník, J., Ikkala, J., Vadakital, V.K.M. and Jääskeläinen, P.O. (2022) 'Real-time rendering of point clouds with photorealistic effects: a survey', *IEEE Access*, Vol. 10, pp.13151–13173.

Kuang, Z., Zhang, Y., Yu, H-X., Agarwala, S., Wu, E. and Wu, J. (2023) 'Stanford-orb: a real-world 3d object inverse rendering benchmark', *Advances in Neural Information Processing Systems*, Vol. 36, pp.46938–46957.

Kumar, Y., Koul, A. and Singh, C. (2023) 'A deep learning approaches in text-to-speech system: a systematic review and recent research perspective', *Multimedia Tools and Applications*, Vol. 82, No. 10, pp.15171–15197.

Lamotte, C. (2022) 'Discovering animation manuals: their place and role in the history of animation', *Animation*, Vol. 17, No. 1, pp.127–143.

Lee, R., Venieris, S.I. and Lane, N.D. (2021) 'Deep neural network-based enhancement for image and video streaming systems: a survey and future directions', *ACM Computing Surveys (CSUR)*, Vol. 54, No. 8, pp.1–30.

Li, L., Goh, T-T. and Jin, D. (2020) 'How textual quality of online reviews affect classification performance: a case of deep learning sentiment analysis', *Neural Computing and Applications*, Vol. 32, pp.4387–4415.

Liu, C., Wang, L., Li, Z., Quan, S. and Xu, Y. (2022) 'Real-time lighting estimation for augmented reality via differentiable screen-space rendering', *IEEE Transactions on Visualization and Computer Graphics*, Vol. 29, No. 4, pp.2132–2145.

Liu, Y., Pu, H. and Sun, D-W. (2021) 'Efficient extraction of deep image features using convolutional neural network (CNN) for applications in detecting and analysing complex food matrices', *Trends in Food Science & Technology*, Vol. 113, pp.193–204.

Marques, B.A.D., Clua, E.W.G., Montenegro, A.A. and Vasconcelos, C.N. (2022) 'Spatially and color consistent environment lighting estimation using deep neural networks for mixed reality', *Computers & Graphics*, Vol. 102, pp.257–268.

Niu, M. and Lo, C-H. (2022) 'Do we see rendered surface materials differently in virtual reality? A psychophysics-based investigation', *Virtual Reality*, Vol. 26, No. 3, pp.1031–1045.

Pandey, S., Yazdanbakhsh, A. and Liu, H. (2024) 'TAO: re-thinking DL-based microarchitecture simulation', *Proceedings of the ACM on Measurement and Analysis of Computing Systems*, Vol. 8, No. 2, pp.1–25.

Pattanayak, S.K., Bhoyar, M. and Adimulam, T. (2024) 'Deep reinforcement learning for complex decision-making tasks', *International Journal of Innovative Research in Science, Engineering and Technology*, Vol. 13, No. 11, pp.18205–18220.

Pontén, O., Xiao, L., Kutter, J., Cui, Y., Wählby, C. and Behrendt, L. (2024) 'PACMan: a software package for automated single-cell chlorophyll fluorometry', *Cytometry Part A*, Vol. 105, No. 3, pp.203–213.

Sajjanar, R., Dixit, U.D. and Vagga, V.K. (2024) 'Advancements in hybrid approaches for brain tumor segmentation in MRI: a comprehensive review of machine learning and deep learning techniques', *Multimedia Tools and Applications*, Vol. 83, No. 10, pp.30505–30539.

Wang, H-W. (2025) 'Application methods of deep learning in enhancing visual effects in film and television post-production', *Journal of Computers*, Vol. 36, No. 1, pp.191–204.

Zuo, C., Qian, J., Feng, S., Yin, W., Li, Y., Fan, P., Han, J., Qian, K. and Chen, Q. (2022) 'Deep learning in optical metrology: a review', *Light: Science & Applications*, Vol. 11, No. 1, pp.1–54.