Proceedings of the 5th International Conference on Computing and Data Science
DOI: 10.54254/2755-2721/16/20230899

Research on 3D scene graph rendering based on neural radiance fields

Hongyang Cui

School of Computer Science and Engineering, Sun Yat-sen University, Guangzhou, 510000, China

Cuihy27@mail2.sysu.edu.cn

Abstract. Since the release of Neural Radiance Fields (NeRF), it has become the research focus of computer graphics in only three years and has become a representative method of using deep learning to deal with computer graphics tasks. The application scenarios of NeRF are very extensive, including virtual reality, augmented reality, computer graphics, 3D scene graph rendering, and other fields. However, the literatures on these aspects are not sufficient. To this end, this article introduces the basic principles of NeRF and compares and analyzes the advantages and disadvantages of NeRF and traditional graphics methods in 3D rendering tasks. According to different improvement directions, from the perspective of improving training speed, improving network generalization, and expanding to dynamic scenes, etc. Representative follow-up work based on NeRF is summarized. And the future application scenarios and development direction of NeRF are prospected.

Keywords: 3D scene graph, rendering, NeRF.

1. Introduction

Compared to computer vision, especially compared to computer vision based on deep learning, computer graphics is more difficult and obscure. There are countless computer vision tasks swept by deep learning, but there are still few computer graphics tasks swept by deep learning. However, the release of Neural Radiance Fields (NeRF) on ECCV in 2020 set off a wave of deep learning methods for computer graphics. In the past less than 2 years, the number of papers on NeRF has been considerable. The application scenarios of NeRF are very extensive, including virtual reality, augmented reality, computer graphics, robot vision, and other fields [1]. For example, it can be used to create photorealistic virtual reality experiences, generate realistic 3D models, provide scene awareness for robots, and more.

Since NeRF and its many follow-up works have given excellent results on very important rendering tasks in graphics, it can be predicted that the work of using deep learning to complete graphics tasks will grow rapidly in the future. At present, NeRF research is still developing, and researchers have proposed many methods to improve NeRF performance and efficiency, such as NeRF++, NSVF, etc. In addition, there are many studies related to NeRF, such as research on view consistency, view sampling, deep learning, and rendering.

This article mainly focuses on the theme of 3D rendering, starting from the principle of traditional computer graphics rendering, introducing the basis of NeRF being proposed, and then introducing the basic principles of NeRF and the training and rendering process, and comparing it with the traditional

DOI: 10.54254/2755-2721/16/20230899

rendering method mentioned above, stating the advantages and disadvantages of NeRF on 3D rendering tasks. Finally, some works that have achieved outstanding results in improving the shortcomings of NeRF are listed, and a wider application field including NeRF is introduced.

2. Related surveys and course notes

2.1. Traditional graphics scene representation methods

2.1.1. Representation of planes. **Point Clouds.**A point cloud is a collection of a large number of discrete three-dimensional points (x,y,z), each point contains its position information in three-dimensional space and possible attribute information, such as color, normal vector, etc. A surface in space can be represented discretized by a point cloud. If you combine the point cloud with the normal vector, specify a normal vector for each point in the point cloud to indicate the vertical direction pointing to the surface of the point, you can get the directional point cloud (surfels), the directional point cloud can help improve the point cloud Processing efficiency and accuracy [2]. For example, in 3D reconstruction and model fitting, oriented point clouds can help identify the curvature and shape of a surface to better fit a model or generate a surface mesh (Figure 1). In addition, oriented point clouds can also be used to improve the rendering effect and make the rendering more realistic, because the oriented point cloud can help the ray tracer to more accurately calculate the normal vector of the intersection point of the ray and the surface.

Figure 1. Point cloud 3D scene.

Grid Meshes. Another way to represent a three-dimensional plane is a grid. The grid is composed of a series of triangular faces, each triangular face contains three vertices, and their connections form a triangle. Grids can be broadly classified into structured grids and unstructured grids. In a structured grid, nodes and edges are connected in a very regular way, each node has a fixed number of adjacent nodes, and the connecting lines also intersect at right angles. This grid has certain advantages in calculation accuracy and calculation efficiency. In an unstructured grid, the connection mode of nodes and edges is irregular, the number of adjacent nodes of each node can be different, and the connection lines can also cross and connect arbitrarily (Figure 2). This mesh is more flexible and convenient when dealing with complex geometries and boundary conditions. Most of the current traditional graphics methods are based on triangular meshes.

DOI: 10.54254/2755-2721/16/20230899

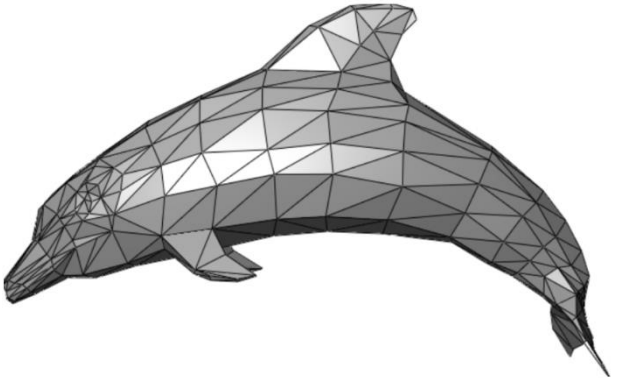

Figure 2. Grid 3D scene.

2.2. Traditional graphics rendering methods

2.2.1. Raster rendering. Raster rendering is the most traditional computer graphics rendering technology. Its basic idea is to perform geometric processing first, project objects in a three-dimensional scene onto a two-dimensional plane according to the viewing angle, and convert them into graphics in a two-dimensional plane coordinate system(primitive) (Figure 3). Secondly, rasterization is carried out, the primitives obtained in the previous step are traversed one by one, and the continuous graphics are converted into discrete pixel regions (fragments) using interpolation algorithms. Then perform fragment processing, calculate the color and brightness of each pixel by simulating the propagation and reflection of light in the scene, color each pixel according to the color and brightness information of the pixel, apply texture, shadow, and other effects, and finally form an image. Raster rendering technology has high image quality and good real-time performance, and is widely used in game development, animation production, virtual reality and real-time visualization, and other fields.

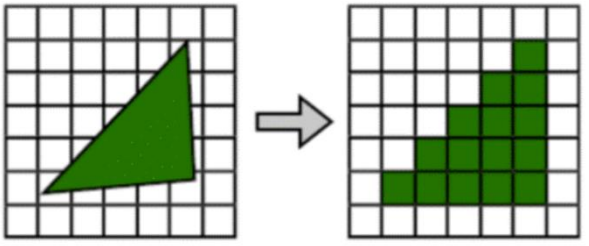

Figure 3. Schematic diagram of the raster rendering process.

2.2.2. Ray tracing rendering. Ray-traced rendering generates images by simulating the propagation and reflection of light through a scene (Figure 4). Different from raster rendering technology, ray tracing technology simulates the light entering the camera from a physical point of view based on the principle of reversible light path, more accurately simulates the propagation and reflection of light, and thus produces a more realistic image effect. In the ray tracing rendering process, the program simulates many rays emitted from the camera position, the rays pass through the objects in the scene, and each collision generates a new ray, which travels in the direction of reflection or refraction, otherwise, continues direct propagation along the original path. The propagation results can be divided into the following situations. If it can reach the light source, calculate the color after reaching the light source. If it cannot reach the light source, calculate the color that cannot reach the light source. Since the light cannot touch the object and is reflected endlessly, we can set a threshold by ourselves, that is, if the number of light reflections exceeds the threshold and has not hit the light source, we judge that it cannot find the light source and stop continuing to reflect. By recursively calculating the intersection point of each ray and object, and

DOI: 10.54254/2755-2721/16/20230899

combining material properties and ray paths to calculate color and brightness information, a high-quality image is finally generated.

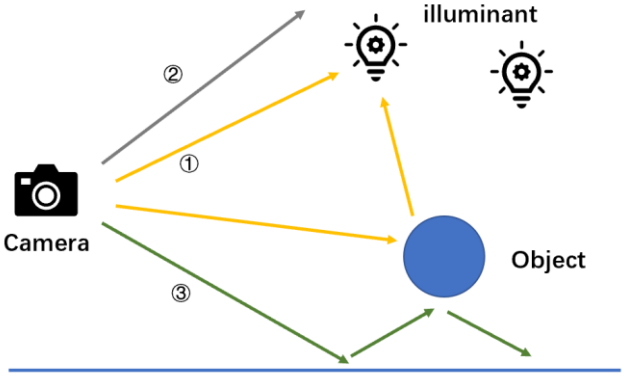

Figure 4. Schematic diagram of ray tracing rendering (① reachable light source, ② unreachable light source, ③ exceeding the threshold as unreachable).

3. The principle and application of NeRF

3.1. Principle of NeRF

NeRF is a neural network-based 3D scene reconstruction technique that predicts the appearance and shape of a scene from a set of input images. The NeRF model represents a 3D scene as a continuous function called a "radiation field" that maps 3D space coordinates to color and opacity.

NeRF works by training a deep neural network to learn this radiation field function. The training dataset includes input images and corresponding 3D scene geometry information. Once trained, the NeRF model can render new views of the scene from any viewpoint and under any lighting condition.

3.1.1. NeRF scene representation method. Using neural networks to train 3D scenes to generate rendering models can be based on a variety of 3D scene representation methods. Many 3D-aware image generation methods based on traditional scene representation methods usually use convolutional architectures. For example, on the point cloud, voxel, and grid, the method of deep learning is directly used to train CNN with 3D filters, but the volume data will become very large, and the processing of 3D CNN will be very slow, so it is necessary to compromise to a lower resolution, will bring the cost of quantization error.

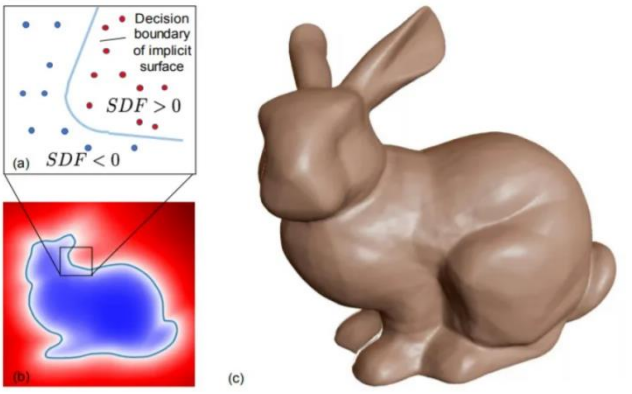

Figure 5. SDF 3D scene representation diagram.

On CVPR 2019, neural implicit scene representation began to appear, that is, the work of using a neural network to fit a scalar function to represent a three-dimensional scene. SDF is the first neural implicit scene representation method. We generally think of multi-layer perceptrons (MLPs) as universal

DOI: 10.54254/2755-2721/16/20230899

function approximators (Figure 5). DeepSDF expresses the three-dimensional surface by regressing an MLP. As shown in the figure 5, the place where SDF>0 means that the point is outside the three-dimensional surface, and the place where SDF<0 means that the point is inside the three-dimensional surface. The neural network that returns this distribution is a multi-layer perceptron, which is a very simple and primitive neural network structure. On this basis, NeRF uses RGB σ for three-dimensional representation. In addition to inferring the distance from the surface of the object like SDF, it can also infer RGB color and transparency. The specific method is as follows: NeRF constructs a function map g_θ, $(\sigma, \boldsymbol{c}) = g_\theta(\boldsymbol{x}, \boldsymbol{d})$, where the input $\boldsymbol{x} \in \mathbb{R}^3$ is a coordinate point in three-dimensional space, $\boldsymbol{d} \in \mathbb{R}^2$ that is, the viewing angle, and the output is $\boldsymbol{\sigma} \in \mathbb{R}^+$ transparency, which $\boldsymbol{c} \in \mathbb{R}^3$ is the color value of RGB (Figure 6).

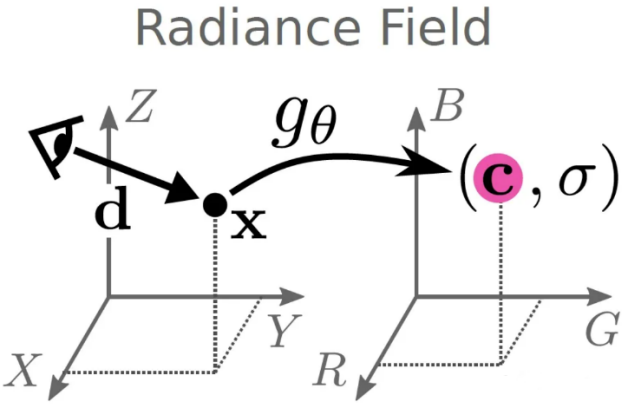

Figure 6. Schematic diagram of the principle of the nerve radiation field [2].

Given the coordinates of a 3D point and the current viewing angle, g_θ the color and transparency corresponding to the 3D point are obtained immediately through mapping to complete the rendering. Therefore, this mapping g_θ capability is an implicit representation of the 3D scene.

3.1.2. NeRF training process. The input data of NeRF are pictures of the same scene taken from different positions and the pose of the camera that took these pictures, the internal parameters of the camera, and the range of the scene. If the image data set lacks the true value of the camera parameters, the classic SfM reconstruction solution COLMAP is used to estimate the required parameters and used as the true value.

The volume rendering function is differentiable, so it can be trained to optimize the NeRF scene representation by minimizing the difference between the synthesized and real images rendered in the previous frame. This is done in the process of training to use NeRF to render new images. First, input these positions into MLP to generate volume density and RGB color values. Then, take different positions. Once this NeRF training is finished, a model that is represented by the multi-layer perceptron's weight is obtained. Each model, of course, only has information about the current scene and is unable to produce images of other scenes.

3.1.3. NeRF rendering process. After obtaining the model trained by NeRF, for the same scene, we only need to input the angle of view and position information to obtain the rendered pixel information of the corresponding position. NeRF's rendering uses the principle of classic volume rendering to solve the color of any light passing through the scene, to render and synthesize a new image. In order to render the neural radiation field according to a certain viewpoint, we first pass the camera ray through the scene, generating a set of 3D sample points; Then let these 3D points and the corresponding 3D viewing direction be used as the input of the neural network to generate a set of colors and densities; Finally, using classic stereoscopic rendering techniques, these colors and densities are added to get a 2D image. Since the above process is derivable, we can use gradient descent to optimize the model, minimize the error between the observed image and the image calculated by model regression, and make the optimized

DOI: 10.54254/2755-2721/16/20230899

model consistent, that is, at the location containing the scene content, we can get larger bulk density, and accurate color.

3.2. Advantages of NeRF

NeRF has the ability to capture complex lighting and shadow effects, so it is very suitable for applications in fields such as computer graphics, virtual reality, and augmented reality. It can be used to create virtual environments, generate realistic 3D models, and even simulate the interaction of light and objects at the microscopic scale. At the same time, the position encoding adopted by it changes the 5D coordinates, which overcomes a key problem of volume representation: when representing complex scenes with high resolution, the storage space cost of discrete voxel grids is very high

3.3. NeRF improvement in rendering

As an epoch-making pioneer of neural network 3D rendering methods, NeRF also has many problems while providing new ideas.

First, when we use the NeRF method to rendering pictures, it takes nearly 200 forward predictions of the MLP depth model to generate one pixel. Although the scale of a single calculation is not large, the amount of calculation required to complete the rendering of the entire image pixel by pixel is still considerable. Second, the training time required by NeRF for each scene is also very slow. For the problem of slow training time, Depth-supervised NeRF uses the sparse output of SfM to supervise NeRF, which can achieve less view input and faster training speed [3].

Second, there is a generalization limit to the NeRF approach. It cannot be expanded to create dynamic scenarios and can only generate one static scene. This problem is mainly combined with monocular video to learn the implicit representation of the scene from monocular video. Using time-varying continuous functions of appearance, geometry, and 3D scene motion, Neural Scene Flow Fields construct dynamic scenes [4]. The sole input needed for the procedure is a monocular video of a recognized camera posture.

Third, the NeRF method has its limit in generalization. It needs to be retrained for a new scene, and cannot be directly extended to unseen scenes. Therefore, some articles began to improve the generalization of NeRF. GrF generates general and detailed point representations by learning local features for each pixel in 2D images and then projecting these features to 3D points [5]. Similar to it are IBRnet, pixelNeRF etc. The core idea is the combination of convolution and NeRF. However, this kind of generalization is still relatively preliminary and cannot achieve ideal results in complex scenes.

Fourth, NeRF requires a large number of pictures from different perspectives for training, which limits its application in reality. pixelNeRF is similar to GRF, using a CNN Encoder to propose image features, so that 3D points are generalizable and support a small amount of input [6]. pixelNeRF can support one image input. From my personal experience, the improvement of NeRF generalization and the number of views is currently limited to a relatively closed test environment, such as synthetic objects or single objects, and the effect in the real open world is not good. CVPR2022 has some open work, such as Urban-NeRF, Block-NeRF, etc., trying to use NeRF to model in complex environments. There are also some works that have improved the NeRF framework, among which Mip-NeRF has a substantial breakthrough. Mip-NeRFproposed a frustum-based sampling strategy, using MPI (Multi-Plane Image) instead of NeRF's RGBσ as the output of the network to achieve NeRF-based anti-aliasing [7]. Mip-NeRF reduces unpleasant aliasing artifacts and significantly improves NeRF's ability to represent fine details while being 7% faster than NeRF and half the size.

3.4. Applications of NeRF in other fields

NeRF has a wide range of application prospects. In addition to achieving outstanding results in the field of 3D rendering, it can also be combined with other tasks.

With the aim of generating new perspectives, altering materials or lighting, or producing new animations, NeRF can assist us in estimating several model parameters (camera, geometry, material, and light parameters) using real data. Self-Calibrating proposes a joint learning of scene geometry and

DOI: 10.54254/2755-2721/16/20230899

precise camera parameters without any calibration object, and proposes a camera self-calibration algorithm suitable for ordinary cameras with arbitrary nonlinear distortion [8]. NeRF is used to model the human body in the field of the digital human body, just like other 3D scene representations. 4D Facial Avatar combines 3DMM and NeRF to implement a dynamic neural radiation field for facial modeling [9]. Animatable introduces neural hybrid weight fields to generate deformation fields, enabling human body modeling [10].

NeRF can also be combined with multimodality, using not only video and images but also text and audio as input. For example, CLIP-NeRF combines CLIP and NeRF to realize editing scenes through text and images [11]. It is currently limited to simple models such as chairs and cars.

NeRF can also play an important role in the field of image and video processing, from the perspective of neural fields to perform image processing such as compression, denoising, super-resolution, inpainting, or for video compression and video editing. In addition, NeRF can also play more roles in the face of robotics, medical imaging, etc.

4. Conclusion

This paper reviews the 3D rendering methods of traditional graphics, explain the basic principles of NeRF, and compare the advantages and disadvantages of NeRF and traditional graphics in 3D rendering tasks. And aiming at the shortcomings of NeRF, from the three perspectives of improving the training speed, improving the generalization of the network, and expanding to dynamic scenarios, it summarizes the optimization methods related to NeRF, and finally summarizes and looks forward to other specific application scenarios of NeRF. It believes that the current NeRF has only just opened the door for deep learning methods to handle graphics applications. Although there are still many problems, many excellent effects in the paper are limited to training samples and cannot be implemented in reality. But with the deepening of more research, we hope that in the near future, computer graphics methods based on deep learning can bring us more exciting visual effects.

References

[1] Tewari, Ayush and Thies, Justus and Mildenha, Advances in Neural rendering 2022, arXiv: 2111.05849v2,

[2] Ben Mildenhall, Pratul Srinivasan, Matthew Tancik, NeRF: Representing Scenes as neural Radiance Fields for View Synthesis, 2020 Euro. Conf. Comp. Vis., 1-10.

[3] Deng K, Liu A, Zhu JY, et al. Depth-supervised nerf: Fewer views and faster training for free. 2021, arXiv preprint arXiv:2107.02791.

[4] Li Z, Niklaus S, Snavely N, et al. Neural scene flow fields for space-time view synthesis of dynamic scenes. 2021, Conf. Comput. Vis. Patt. Rec. 6498-6508.

[5] Trevithick A, Yang B. Grf: Learning a general radiance field for 3d representation and rendering, 2021 Conf. Comp. Vis..15182-15192.

[6] Yu A, Ye V, Tancik M, et al. pixel nerf: Neural radiance fields from one or few images 2021, Conf. Comput. Vis. Patt. Rec. 4578-4587.

[7] Barron JT, Mildenhall B, Tancik M, et al. Mip -nerf: A multiscale representation for anti-aliasing neural radiance fields, 2021 Conf. Comp. Vis. 5855-5864.

[8] Jeong Y, Ahn S, Choy C, et al. Self-calibrating neural radiance fields. 2021, Conf. Comp. Vis. 5846-5854.

[9] Gafni G, Thies J, Zollhofer M, et al. Dynamic neural radiance fields for monocular 4d facial avatar reconstruction 2021, Conf. Comput. Vis. Patt. Rec. 8649-8658.

[10] Peng S, Dong J, Wang Q, et al. Animatable neural radiance fields for modeling dynamic human bodies. 2021, Conf. Comp. Vis. 14314-14323.

[11] Wang C, Chai M, He M, et al. CLIP-NeRF: Text-and-Image Driven Manipulation of Neural Radiance Fields. 2021, arXiv preprint arXiv:2112.05139.