3D reconstruction of plant based on NeRF

Bohuai Wang
School of Electrical and Electronic Engineering
Shanghai Institute of Technology
Shanghai, China
1649414755@qq.com

Xizhong Shen
School of Electrical and Electronic Engineering
Shanghai Institute of Technology
Shanghai, China
xzshen@yeah.net

***Abstract*—At present, aiming at the problems of quality and lack of modeling methods in plant 3D modeling technology, an improved fast 3D plant modeling technology based on Neural radiation field (NeRF) is proposed. Taking the simple plant multi view image as the input, the pose of the multi view image is obtained by using SFM. The plant image and pose information are sent to the improved nerf model for training, and the high-quality plant 3D scene is obtained. In order to improve the speed of 3D modeling, this study proposes an improved method of nerf neural radiation field based on instant NGP to reconstruct 3D plants, and a small fusion MLP based on spherical harmonics and multi-resolution hash coding for training and rendering. Compared with the original nerf training time, this method also achieves good rendering quality.**

Keywords—3D reconstruction, Neural radiation field, New perspective synthesis, Spherical harmonics, hash coding

I. Introduction

With the rapid development of digital technology, the preservation of plant specimen information data in the form of pictures or physical objects has been unable to meet the requirements of plant information visualization. There is a great demand for three-dimensional reconstruction of plants. The three-dimensional scene of plants has a strong application demand in ecological environment construction, urban and rural landscape design, ancient and famous trees protection and so on.

However, there are currently some difficulties in 3D modeling of plants, including a wide variety of plant species, varying shapes, diverse growth patterns, as well as structural irregularities and self similarity, which pose many challenges for accurate 3D modeling. This is mainly reflected in the difficulty of obtaining plant appearance information, as well as the difficulty of fine reconstruction and modeling of plants. This is because the plant morphology is diverse, the crown structure is complex, and occlusion causes information disconnection and loss. Traditional surface modeling methods cannot achieve satisfactory results. The complexity and diversity of plant morphology determine that the effective acquisition and rapid reconstruction of plant morphology pose great challenges.

Traditional 3D reconstruction methods usually rely on images from multiple perspectives or sparse depth information, and then infer 3D scenes through computer vision and geometry algorithms. But these methods are sometimes difficult to deal with complex scenes, and often require a lot of manual annotation or complex preprocessing. Such traditional methods have poor robustness and require high quality images. With the development and integration of neural network and computer graphics, a new 3D modeling method using neural radiation field has gradually emerged.

In contrast, nerf uses deep learning to solve this problem. It is based on neural network and represents the scene by learning the radiation intensity of each 3D point in the scene. Radiation intensity determines the color and brightness seen at this point, so it contains information about materials, lighting, and geometry.

II. Principle Explanation

A. Introduction to NeRF

Nerf is a method used to generate realistic 3D scenes, which combines computer graphics and neural network technology to generate a new perspective image from multiple perspectives and pose parameters, without the need for an intermediate 3D modeling process.

Firstly, when reconstructing a 3D scene of plants, multiple images taken from different perspectives are required. These images can come from any appropriate angle and perspective, preferably as many as possible, to better capture the details and complexity of the scene. The neural radiation field represents a 3D scene as a function of 5 vector values, with the inputs being the 3D position x=(x, y, z) and the 2D viewing direction(θ,Φ), Its output is the emission color c=(r, g, b) and volume density. Use MLP network to approximate this continuous 5D scene representation and optimize its weight parameters, mapping each input 5D coordinate to its corresponding volume density and directional emission color.

Then, We predict the density σ which only as a function of position x by limiting the network, while allowing the RGB color c to be predicted as a function of position and viewing direction. To achieve this, MLP first processes the input 3D coordinate x with 8 fully connected layers (using ReLU activation and 256 channels per layer) and outputs σ and 256 dimensional feature vectors. Then, this feature vector is connected to the observation direction of the camera's rays and passed to another fully connected layer (using ReLU activation function and 128 dimensional channels), which outputs RGB colors related to the view.

The NeRF introduces the concept of volume rendering in graphics, which is used to describe the illumination imaging of points in space. There are four types of interactions between photons and particles in the light: absorption, emission, out -scattering, and in-scattering .It can be described by the simplified volume rendering equation as follows:

$$I(s) = \int_0^s T(t)\sigma(t)C(t)dt + T(s)I_0 (1)$$

The rendering equation expresses the radiation situation of a spatial point in a given direction, which is composed of two parts: self emission and refractive radiation, including scattering function, incident direction.

Modeling ray radiation refers to modeling the color corresponding to that ray. And NeRF is a set of MLP that can approximate the rendering equations above. This is also the working principle of NeRF. Under the NeRF based representation method, the 3D scene is represented as a set of learnable and continuous radiation fields. The 3D scene is represented as a continuous and learnable radiation field:

$$F_\theta(x, d) = [\sigma, c](2)$$

Given the coordinates x and observation direction d of a spatial point, the density value of that point can be solved σ Corresponding color value c. If the color value is predicted and the loss is calculated based on the input image corresponding to the current pose, optimization can be carried out to gradually converge the model. After obtaining the radiation field fitted by the neural network, a new color RGB corresponding to the coordinate x and observation direction d can be rendered based on the implicit radiation field, completing the rendering of the new perspective image. Meanwhile, by incorporating perspective information as input, the 3D scene can contain more information about light reflection and perspective changes.

MLP Network Architecture: The architecture of an MLP network is a method of encoding input positions and observation directions, and then mapping them to output colors. The formula used for position encoding is as follows:

$$\gamma(p) = \left(sin(2^0\pi p), cos(2^0\pi p), \dots, sin(2^{L-1}\pi p), cos(2^{L-1}\pi p)\right)(3)$$

The position encoding function transforms the input x (x, y, z) from low dimensional to high-dimensional R → R^{2L}, and encodes each 3-dimensional sampling point with L taken as 10, resulting in a final dimension of 60. The input direction d (x, y, z) is encoded with L taken as 4, resulting in a final dimension of 24, which corresponds to γ (x) 60 and γ (d) 24.

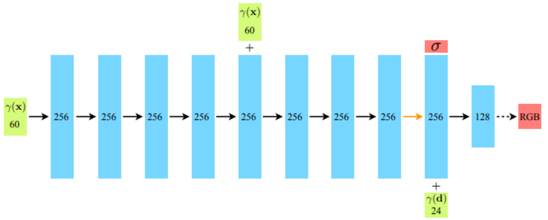

Fig. 1. MLP Network Architecture

As shown in Fig. 1, the neural network input position is γ (x), The dimension of each hidden layer is 256. The position is added in the middle of the neural network to enhance the network representation ability. Finally, the direction information γ (d) is added and the final result is output.

The perspective information includes the origin **o** of the ray (the optical center of the camera), as well as the path of the ray that can be obtained by connecting any point (u, v) on the pixel position with the camera position d:

$$r(t) = o + td(4)$$

Next, sample the 3D points in the scene along the ray direction from the camera position. Based on the volume rendering formula, the color value of the point can be obtained:

$$C(\boldsymbol{r}) = \int_{t_n}^{t_f} T(t)\sigma\big(\boldsymbol{r}(t)\big)\boldsymbol{c}\big(\boldsymbol{r}(t), \boldsymbol{d}\big)dt\ (5)$$

t_n, t_f represents the starting point and focus of integration along the ray, Indicates the transparency of the light at point t, which can be determined by the volume density σ:

$$T(t) = exp\left(-\int_{t_n}^{t_f} \sigma\big(r(s)\big)ds\right)(6)$$

Due to the inability to continuously sample each point to obtain an integral value, its discrete form was introduced, dividing the interval and then sampling it:

$$C(r) = \sum_{i=1}^{N} T_i\big(1 - exp(-\sigma_i\delta_i)\big)c_i(7)$$

The training objective of the model is to minimize the difference between the predicted image and the real image, and the mean square loss function is generally used for optimization. C_c represents rough uniform sampling in the first stage, C_f represents the second fine sampling, and the total mean square loss function is as follows:

$$\mathcal{L} = \sum_{\boldsymbol{r}\in\mathcal{R}} \left[\parallel \hat{C}_c(\boldsymbol{r}) - C(\boldsymbol{r}) \parallel_2^2 + \parallel \hat{C}_f(\boldsymbol{r}) - C(\boldsymbol{r}) \parallel_2^2\right](8)$$

B. Improvement of NeRF

The core idea behind improving the selection of sampling points is to skip sampling in blank space as well as sampling behind high-density areas. These grids roughly mark empty and not empty spaces. If the occupancy rate of the light is too low, the sampling will be skipped. These occupy the grid for independent storage and are updated throughout the training process based on updated density predictions. The Instant NGP method can increase the sampling speed by 10-100 times compared to the original NeRF.

In the traditional NeRF process, a position encoding function is used to map input coordinates to a higher dimensional space. Instant NGP proposes a trainable hash encoding. The idea is to map coordinates to trainable feature vectors, which can be optimized in the standard process of NeRF training. The principle is shown in Fig. 2.

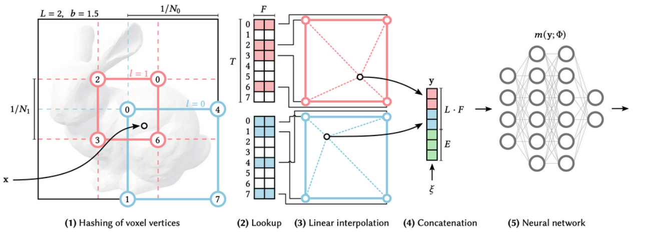

Fig. 2. Hash Encoding Process. Assign an index to the angle of input coordinates for i and hash the coordinates. Then, it searches for vectors corresponding to these indexes in a large feature hash table and searches for specific voxels based on input coordinates. Next, it generates encoded inputs and auxiliary inputs for MLP. During the training process, the loss gradient will propagate back through MLP, cascading, and linear interpolation, and ultimately accumulate in the searched feature vectors for model optimization.

One important advantage of spherical harmonics in rendering is their invariance at different viewing angles. This means that when the observer's viewpoint changes, the spherical harmonic function can adapt to the new perspective in a very efficient and accurate way, without the need to recalculate a large amount of lighting or environmental mapping data. By utilizing spherical harmonics, NeRF rendering can better handle lighting and environmental effects from different viewpoints, thereby improving rendering efficiency and visual quality. The Fig.3 shows the visualization of the basic function of spherical harmonics.

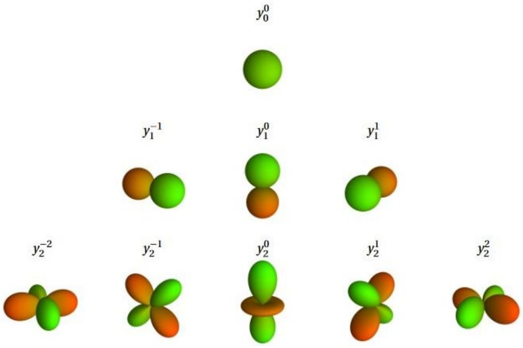

Fig. 3. Spherical harmonic

When rendering the target view of the scene, the camera will emit a camera ray for each pixel and query the scene at points along that ray. We can use different samplers to choose where to query these points. These samplers have some boundary concepts that define the starting and ending points of light rays. If you know that everything in the scene exists within a predefined range (such as a cube that a room can accommodate), the sampler will correctly sample the entire space. However, if the scene is unbounded (i.e. outdoor), defining where to stop requires the use of the following segmented sampler function.

$$f(x)=\begin{cases} x & ||x|| \le 1 \\ \left(2-\frac{1}{||x||}\right)\left(\frac{x}{||x||}\right) & ||x|| > 1 \end{cases} \tag{9}$$

This segmented sampling function is used to generate the initial sampling sample set of the scene. This sampler evenly distributes half of the sampled samples to the position 1 away from the camera. The remaining samples scale as the step size increases. The selection of step size makes the visual cone its own scaled version. By increasing the step size, we can sample distant objects while still having a dense set of samples for nearby objects.

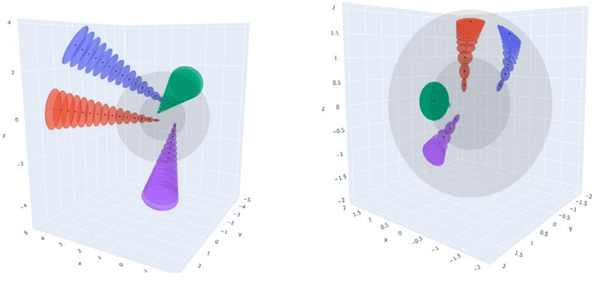

Fig. 4. Sample point shrinkage

We visualize a unit sphere before shrinking the scene, as shown on the left in Fig.4, Visualize a sphere with a radius of 2 after shrinking the scene, as shown on the right in Fig. 4.

III. Experimental methods

Image collection uses a regular mobile phone lens to capture images. For the convenience of the subsequent training process and the acquisition of camera pose, the shooting is directly recorded as a video and then sampled as an image using a script program.

A. Data preprocessing

We need to use COLMAP for data preprocessing to obtain the shooting pose of the input image. First, we used our phones to take the following picture material, as shown in Fig. 5.

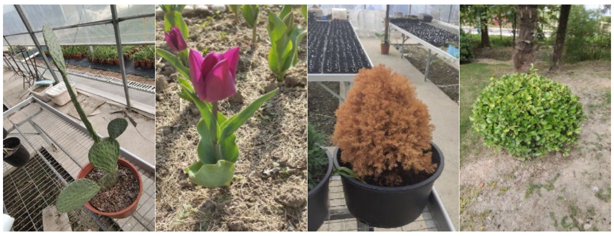

Fig. 5. Plant images taken with phones

During the process of taking plant images, we obtained basic information about the images, including pixels, image size, camera focal length, etc. However, for the subsequent training of the neural radiation field network, we also need to transform the input data (camera origin position and sampled rays for each pixel) into the same coordinate system, And project it into a normalized device coordinate space (NDC Normalized Device Coordinates). This article uses the SFM method provided by COLMAP to estimate the image pose. Fig.6 shows the result of processing the photos in Fig. 5 with COLMAP.

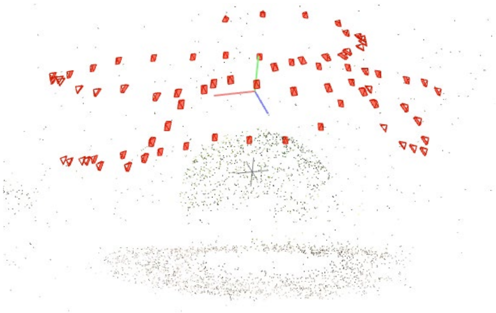

Fig. 6. Colmap generated point cloud map

In the above Fig. 6, the shooting scene is represented by discrete point clouds, and the red part is the calculated virtual camera position.

B. Parameter settings

Using NVIDIA GeForce RTX 3050 GPU for training, the maximum number of iterations is set to 30000 steps, the basic MLP hidden layer is set to 2, the dimension is set to 64, the resolution layer L of hash encoding is 16, the length T of the basic grid of hash encoding is 2^4~2^{10}, the basic resolution of MLP hash encoding is 16, and the feature dimension F of each resolution level is 2, Use the Adam optimizer to dynamically adjust the learning rate, with an initial learning rate set to 0.01. Based on the training stage, use the cosine function to control the decay of the learning rate.

C. Evaluating indicator

Mean Square Error(MSE) is a widely used evaluation indicator for measuring reconstruction quality, commonly used in the field of 3D reconstruction to measure the difference between reconstruction results and real data. MSE calculates the square of the difference between the predicted and true values of each data point, and then takes the average of these square differences.

$$R_{MSE} = \frac{1}{mn}\sum_{i=0}^{m-1}\sum_{j=0}^{n-1}[I(i,j) - K(i,j)]^2 \quad (10)$$

Although MSE is a commonly used evaluation metric, it may not fully capture issues in certain application scenarios. For example, in image processing, MSE tends to excessively penalize high-frequency noise, so in some cases, combining other evaluation indicators can more objectively evaluate experimental results.

Peak Signal to Noise Ratio (PSNR) is a mathematical based image quality evaluation metric commonly used to compare the quality of compressed images with the original image. The underlying idea is that if two images are very similar, the mean square error (MSE) between them should be small, resulting in a high PSNR value.

$$R_{\mathrm{PSNR}} = 10\log_{10}\left(\frac{H_{\max}^2}{R_{\mathrm{MSE}}}\right) (11)$$

The Structural Similarity Index (SSIM) is an evaluation metric used to measure image quality, aimed at evaluating the perceptual quality of images. Compared with mean square error (MSE), SSIM takes into account the brightness, contrast, and structural information of images more comprehensively, and is more suitable for image quality evaluation in some cases.

$$SSIM(x,y) = [I(x,y)]^{\alpha}[c(x,y)]^{\beta}[s(x,y)]^{\gamma} (12)$$

Learning Perceptual Image Patch Similarity (LPIPS), also known as perceptual loss, is used to measure the difference between two images. LPIPS is learned through deep learning models, aiming to more accurately simulate the image similarity perceived by human vision. The lower the value of LPIPS, the more similar the two images are, and vice versa, the greater the difference.

IV. Experimental results and analysis

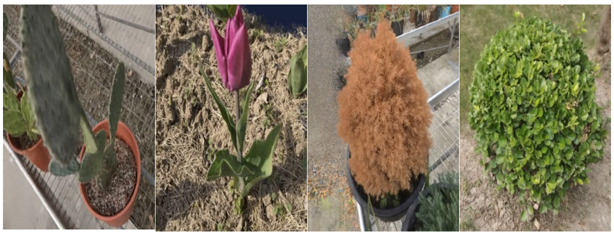

Fig. 7. Rendered plant images

The Fig. 7 shows the results of rendering, and the TABLE I shows the results of various evaluation indicators.

TABLE I. Experimental results of plant scenes

Plant Name	Experimental Indicator		
	PSNR/dB	***SSIM***	***LPIPS***
Cactus	26.76	0.778	0.153
Flower	25.45	0.457	0.252
Pot	22.18	0.687	0.261
Shrub	22.70	0.807	0.250

In TABLE I, we report PSNR, SSIM, and LPIPS across the test images in our dataset. The experimental results in Fig.7 show that the plant scene trained in the radiation field still has a realistic photo effect. After 21-23 minutes of training, the PSNR index of the modeled images of the improved radiation field reached around 25dB, indicating good modeling quality and meeting performance requirements. Although the modeled plants contain complex geometric structures, the parameter indicators of the modeling results have not fluctuated, indicating that the method has good robustness. The color structure of the plants did not show any distortion and was fully restored.

V. Conclusion

NeRF can complete high-quality 3D plant modeling tasks in a short period of time. It only requires collecting multi view images of static clothing, using COLMAP technology to calculate pose, and inputting it into the radiation field for about 20 minutes of training. Practitioners do not need to rely on hardware acquisition equipment and professional software. Compared to traditional 3D vision methods, using radiation field modeling is more robust and adaptable to various plants. In addition, using this method can avoid the phenomenon of model distortion in plant scenes. The research technology achievements can be used in the field of landscape design, as well as in the three-dimensional display of plants in the field of landscape architecture, the establishment of large-scale plant three-dimensional model databases, and the application research of plant scene display.

References

[1] Mildenhall B, Srinivasan P P, Tancik M, et al. Nerf: Representingsce-nes as neural radiance fields for view synthesis[J]. Communications of the ACM, 2021, 65(1): 99-106.

[2] Müller T, Evans A, Schied C, et al. Instant neural graphics primitives with a multiresolution hash encoding[J]. ACM Transactions on Graphics (ToG), 2022, 41(4): 1-15.

[3] Barron J T, Mildenhall B, Tancik M, et al. Mip-nerf: A multiscale representation for anti-aliasing neural radiance fields[C]//Proceedings of the IEEE/CVF International Conference on Computer Vision. 2021: 5855-5864.

[4] Barron J T, Mildenhall B, Verbin D, et al. Mip-nerf 360: Unbounded anti-aliased neural radiance fields[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2022: 5470-5479.

[5] Fridovich-Keil S, Yu A, Tancik M, et al. Plenoxels: Radiance fields without neural networks[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2022: 5501-5510.

[6] Tancik M, Weber E, Ng E, et al. Nerfstudio: A modular framework for neural radiance field development[C]//ACM SIGGRAPH 2023 Conference Proceedings. 2023: 1-12.