Neural Rendering of Underwater Scenes under Flat Refractive Geometry

Ting Bai[a], Xiaoqiang Zhang*[a], Caiyu Xiong[a], Zhixin Zhang[a], Lingyan Ran[b], Hongyu Chu[a], Hu Deng[a]
[a]School of Information Engineering, Southwest University of Science and Technology, Mianyang 621010, China; [b]School of Computer Science, Northwestern Polytechnical University, Xi'an 710072, China

ABSTRACT

As a promising technique, the neural radiance fields (NeRF) and neural rendering are now widely applied in novel view synthesis and scene reconstruction. In the vanilla NeRF and subsequent neural rendering methods, one important assumption of the scene is that there is only one type of light medium in the scene, hence the light rays in such methods would remain straight during rendering. However, for underwater scenes, the camera is usually placed in a water-proofing and transplant housing. The light ray path in such a scenario would be air-water or air-housing material-water, which would cause refraction and violate the basic assumptions in vanilla NeRF. To address the issue of novel view synthesis in scenes with refractive media, this paper proposes a refractive neural rendering method under flat refractive geometry. First, the distance from the origin of the light ray to the refraction plane and the normal vector are pre-calibrated, which is utilized to model the per-pixel refracted ray direction and the intersection point of the refraction plane. With the refracted rays, a neural radiance field can be trained and can be used for novel view synthesis in refractive scenes. The method is validated on synthetic and real data, revealing accurate novel view synthesis of scenes under refractive surfaces from sparse multi-view images.

Keywords: Novel view synthesis, Flat Refractive Geometry, Neural Rendering

INTRODUCTION

With the wide application of underwater detection equipment such as underwater vehicles and unmanned underwater vehicles, underwater optical vision systems have developed considerably [1]. View synthesis leveraging underwater 2D images represents a pivotal research trajectory within computer vision and graphics, promising significant advancements in the field. As a novel implicit scene neural rendering method, Neural Radiance Fields (NeRF)[2] possess the capability to reconstruct intricate 3D scenes with exceptional precision, solely from a sparse collection of 2D images. Its excellent visual effects have attracted the attention of researchers. However, NeRF and its successors mostly assume that there is only one type of medium in the scene, such as air, which can sample light along a straight line. For the refraction of light when there is more than one medium in the scene, such as underwater scenes, this refracted scene no longer conforms to NeRF's linear sampling assumption, and the scene will be caused by unnecessary deformation.

The continuous development of deep learning technology also provides new ideas for NeRF in complex light ray scenes. LB-NeRF[3] adds a continuous volume deformation layer to model ray bending. It took the refraction effect of light into account by adding an offset to the straight line of light emitted by the camera. However, the offset is not physically fixed and cannot model ray refraction. NeReF[4] recovers a normal volume field and the corresponding density field. Each ray will accumulate the refraction points on the path and the corresponding normal vector for refraction during the query. Although the ray is not a straight line, each refracted ray is unique. NeRFrac[5] leverages an MLP-based refraction field to approximate ray-surface distance, subsequently computing refracted rays at intersections using estimated normal and incident rays.

Considering the influence of refraction, this paper proposed a neural network rendering method for underwater scenes based on flat refraction geometry. In this paper, the flat refractive geometry model represents a streamlined and simplified approach to modeling., the light from a plane passing through the homogeneous medium, refraction to another homogeneous medium with a different refractive index. Instead of simply adding an offset to the ray, we first estimate the distance and normal from the origin of the ray to the refraction plane by casing parameter calibration. According to the NeRF of simulating ray radiation, it is easy to get the intersection point between the ray and the refracting plane, and

Fifteenth International Conference on Information Optics and Photonics
(CIOP 2024), edited by Yue Yang, Proc. of SPIE Vol. 13418, 134182G
 · 0277-786X · doi: 10.1117/12.3048562

then calculate the refracting ray. After that, the neural radiation field of the underwater scene is trained and rendered to synthesize the new view. We validate and evaluate the viability and efficacy of the introduced methodology in underwater scenarios. We built new synthetic datasets to create underwater scenes using Taichi's ray tracing[6]. A dataset of new real underwater scenes is obtained from array sub-views of a light field camera, where objects are placed in a flat glass tank and the camera is positioned to capture images from a vantage point external to the water's surface.

RELATED WORK

Flat refraction and parameter calibration

In underwater scenes, the imaging sensor is usually placed in a transparent waterproof shell, and the refractive index of air, shell transparent material, and water are different, so the light path will be refracted at the interface between the inner and outer shell, which is simulated as a flat refractive model[7]. Underwater calibration of 2D imaging cameras has been studied based on planar refractive geometry. Agrawal et al.[7] used the proposed geometric constraint of planar refraction to estimate the camera housing parameters for scenes with different media. Chen et al.[8] proposed a linear method to estimate the thickness of the medium using the normal vector of the refracting surface and used the spherical search method to estimate the normal vector of the best refracting surface. Luczynski et al.[9] combined the virtual pinhole camera model with the projection function of the axial camera model, used the Pinax model to compensate for the image distortion caused by refraction, and performed fast refraction correction on the plane. Kang et al.[10] the robust rotation SAM algorithm was introduced into a new hybrid optimization framework, and the local sliding (LS) method was used based on beam adjustment for dual-look underwater calibration. Pedersen et al.[11] combined with ray tracing, used Snell's law to simulate light refraction and compensate for refraction errors to realize underwater calibration. Zhang et al.[12] modeled the scene light path and camera light path of the light field camera in underwater scenes based on the multiple projection center model and calibrated the underwater parameters of the light field camera by leveraging geometric constraints of flat refraction.

NeRF and neural rendering

Neural rendering is an important area where machine learning techniques and computer graphics are combined. Typical neural rendering takes multi-view images and videos of some scenes as input, uses a "neural" scene representation, and "renders" new view images of the scene. NeRF[2] uses fully connected multi-layer perception (MLP) to represent the entire scene as a continuous volumetric light field. Specifically, NeRF learns density values and color values at each location in space and then uses volume rendering to create an image of a scene as seen from any given viewpoint. As a new viewpoint synthesis and 3D reconstruction method, NeRF has also attracted much attention in various fields and has made great progress in various directions.

MIP-Nerf[13] transforms sampled rays into anti-aliased conical frustums to eliminate blur and overlapping rendering, and MIP-Nerf360[14] renders the scene in a 360° surrounding the target point based on MIP-Nerf. Tensorf[15] represents radiance fields as 4D tensors and decomposes 4D tensors into compact low-order tensors, making it perform better both training efficiency and the quality of rendered output. IBRNet[16] constructs a viewpoint interpolation function, and ray transformation and MLP constitute the core network framework for synthesizing new viewpoints of complex scenes. WaterNeRF[17] estimated the parameters of a physics-based underwater imaging model to generate corrected underwater images after determining the scene structure and radiance fields. Beyond NeRF[18] simulates the effect of water by jointly learning the medium and neural radiance field to learn the refractive index and volume density of the underwater environment, to recover the true color effectively. Novel view synthesis refers to the process of creating an image of a scene from a unique perspective that has not been previously captured or observed within the input images of that particular scene. NeRF has always assumed that light is sampled along a straight line in a homogeneous medium, and in some cases, the presence of light divergence between different views may lead to distortion when generating new viewpoints. NeRFReN[19] addresses the complex distortion problem by splitting the scene into different components and models it using a separate neural radiance field. Deblur-Nerf[20] employs the deformable sparse kernel approach to refine ray paths, extremely well mimicking the process of image blurring and thereby achieving optimized results. Nerfies[21] and LB-Nerf[3] use continuous volume deformation fields, which can be interpolated in space and time to model the bending of light in transparent media. However, the above methods cannot completely solve the refraction problem that may occur in the scene. Influenced by these studies, we introduce a planar refraction geometry model to simulate refraction before the neural rendering of underwater scenes.

METHODOLOGY

Figure 1 shows our overall pipeline of this paper. We model the refraction scene as a flat refractive geometry model and use the off-line calibration method to obtain the refraction-related parameters. The intersections of light rays with the refracting surface and the refracting light ray can be easily obtained, and achieve high-quality rendering of refracting scenes and synthesis of novel views by training a NeRF. In all scenes in this paper, the camera is directly in front of the refracting surface, which belongs to the forward setting[22].

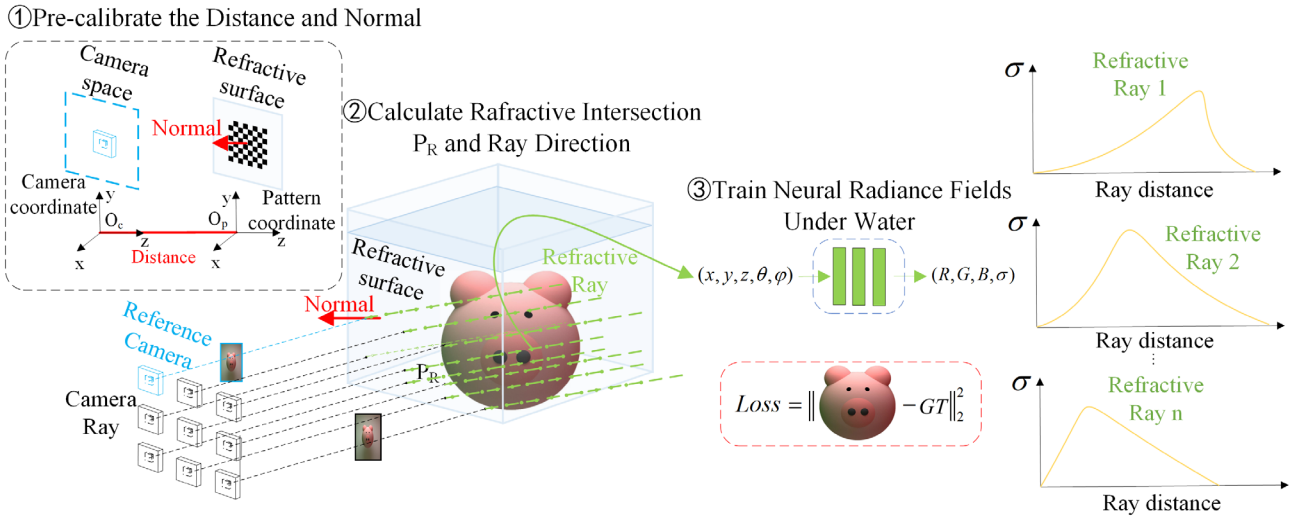

Figure 1. Our method consists of three steps: ① Pre-calibrate the normal to the refractive surface and the distance from the reference camera to the refractive surface using the calibration pattern. ② Calculate the refractive intersection point and direction of the original camera light through the refractive surface with normal distance. ③ Training neural radiance fields whose input is obtained mathematical model of refractive rays consisting of refracted intersection points and direction. A simple image photometric loss is used to train our method.

Flat refractive geometry model and parameter calibration

As shown in Figure 2, this is a general forward refraction model of flat refractive geometry, where the camera observes a known target object through parallel planar refracting layers. We assume that in the camera coordinate system, the camera center is located at the origin. Let n denote the coplanar normal of each layer, $[d_i, \lambda_i]$ denote the thickness and refractive index of the ith layer medium, respectively, and d_0 denote the distance between the camera and the refracting plane of the first layer. There is a target 3D point in the rightmost medium λ_i, and the light from this point is refracted by n different refracting planes and enters the camera on the left. This model is also applicable to back-refracted projections.

All refraction in nature follows Snell's law, which is a fundamental law in optics that describes the angular relationship between the refraction of light as it travels between two different media:

$$\lambda_1 \sin(\theta_1) = \lambda_2 \sin(\theta_2), \tag{1}$$

where λ_1, λ_2 are the refractive index of two different media, and θ_1, θ_2 are the incidence and refraction angles. It is not the most intuitive way to describe the angle of incidence and refraction of light rays in 3D space, and a vector expression using Snell's law is more appropriate. Suppose that in two adjacent dielectric layers, $I \in \mathcal{R}^3$ is the incident ray, $N \in \mathcal{R}^3$ is the normal vector, and $T \in \mathcal{R}^3$ is the refracted ray:

$$T = \left(\frac{\lambda_1}{\lambda_2} N \cdot I - \sqrt{1 - \frac{\lambda_1^2}{\lambda_2^2}[1 - (N \cdot I)^2]} \right) N + \frac{\lambda_1}{\lambda_2} N. \tag{2}$$

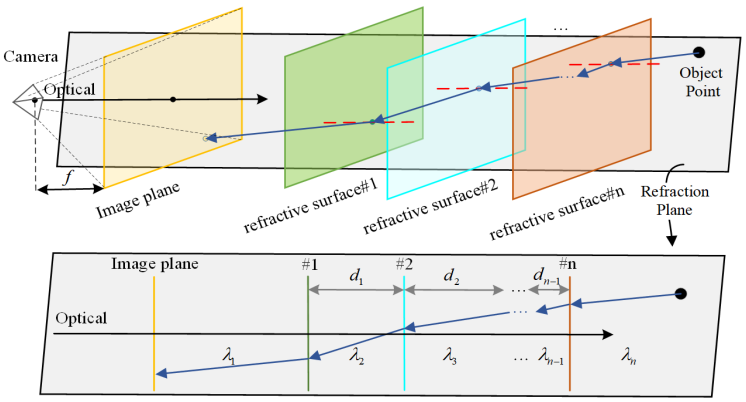

Figure 2. Schematic diagram of the multi-layer flat refraction model in camera imaging

To obtain the refraction ray T and completely describe the refraction scene light path, it is necessary to calibrate some unknown parameters in the plane refraction model, which mainly include the normal vector N of the refraction plane and the distance d_0 from the camera to the refraction plane. According to Zhang's calibration[23], the image of the planar checkerboard calibration board located on the refractivity plane #1 is used to estimate the pose and position of the camera perspective concerning the calibration board. Based on this, the true values of N and d_0 can be further calculated. This method of obtaining the true value of N and d_0 through the checkerboard calibration board on the refraction surface is also mentioned in the literature[12].

We first define the interface between two different media as the interface, which is the refracting surface. The origin of the interface coordinate system is at the ray incident point, the X and Y axes are located inside the interface, the Z axis is perpendicular to the interface, and the positive direction points to the medium where the ray is incident. We set the plane normal vector in the interface coordinate system as $n=[0,0,-1]^T$, and the plane normal vector $N=R\cdot n$ in the camera coordinate system can be obtained by transforming the rotation matrix R. The optical center in the interface coordinate system is represented as $O=-R^{-1}\cdot t$ by the translation vector t, and the distance from the camera to the refracting plane is represented by the absolute value of the z-axis coordinate of the optical center in the interface coordinate system $d_0=|O(3)|$.

Neural radiance Fields in a refracting medium

In scenes involving refraction phenomena, vanilla NeRF needs to be adjusted to account for refraction when performing the volume rendering process. This means that when sampling the ray path, it is necessary to calculate the refraction point S of the ray at the interface and adjust the position and weight of the sampling points accordingly to improve the rendering efficiency. The batches of rays are calculated by determining the ray in the cone corresponding to each pixel based on the camera parameters:

$$r(t)=o+tv\,, \tag{3}$$

Where 0 is the 3D origin position and v is the direction vector. The output depth of each ray to the refracting surface D :

$$D=\frac{d_0}{\frac{-N\cdot v}{|N||v|}}\,. \tag{4}$$

From this, the intersection $r(D)=o+Dv$ between the input ray and the refracted surface is further calculated. $r(D)$ is the refraction point S of the light at the boundary of the medium. When the ray r reaches the refraction point S, it will be re-fracted. According to the normal vector of the refraction surface that has been calculated, the refracted ray r' is obtained by using Snell's law. The vertical relation and vector form of Snell's law are given in Equations (1) and (2). Since the intermediate medium of the real experiment is a transparent waterproof outer cover, whose thickness can be

measured in advance [5], and the interface is parallel, the refracted light in water can be easily obtained through Equations (1) and (2).

The ray r' in the refracting plane is given by:

$$r'(t) = s + tv' , \tag{5}$$

Once the refracted ray r' is available, each pixel position of the underwater scene image can be rendered. Sampling along the refracted ray takes the spatial sampling point's 3D coordinate position and direction as input. x is first fed to the MLP network and outputs the corresponding voxel density σ and 256-dimensional intermediate features. The intermediate feature, along with orientation (θ, φ), where θ represents polar angle and φ represents the azimuthal angle. They are fed into an additional 128-dimensional fully connected layer to predict the color value $c = (r, g, b)$ associated with the orientation of that 3D point. The color of this refracted ray under the water is expressed as the integral:

$$C(r) = \int_{t_n}^{t_f} T(t) \cdot \sigma(r'(t)) \cdot c(r'(t), d) dt , \tag{6}$$

where $T(t)$ represents the cumulative transparency of the ray from t_n to t, t_n and t_f are the near and far bounds, respecti -vely, and they are written as follows:

$$T(t) = \exp(-\int_{t_n}^{t} \sigma(r'(t)) ds) , \tag{7}$$

The rendering loss is calculated by comparing the final render to the ground truth color of each ray:

$$L = \sum_{r \in R} \left\| C(r) - C_{GT}(r) \right\|_2^2 . \tag{8}$$

EXPERIMENTAL RESULTS

Dataset and equipment

To verify the feasibility and effectiveness of the proposed method, we build four synthetic datasets and four groups of real datasets for experiments. Where we assume $\lambda_{air} = 1$ and $\lambda_{water} = 1.33$, and the experiments focused on the synthesis of new viewpoint images.

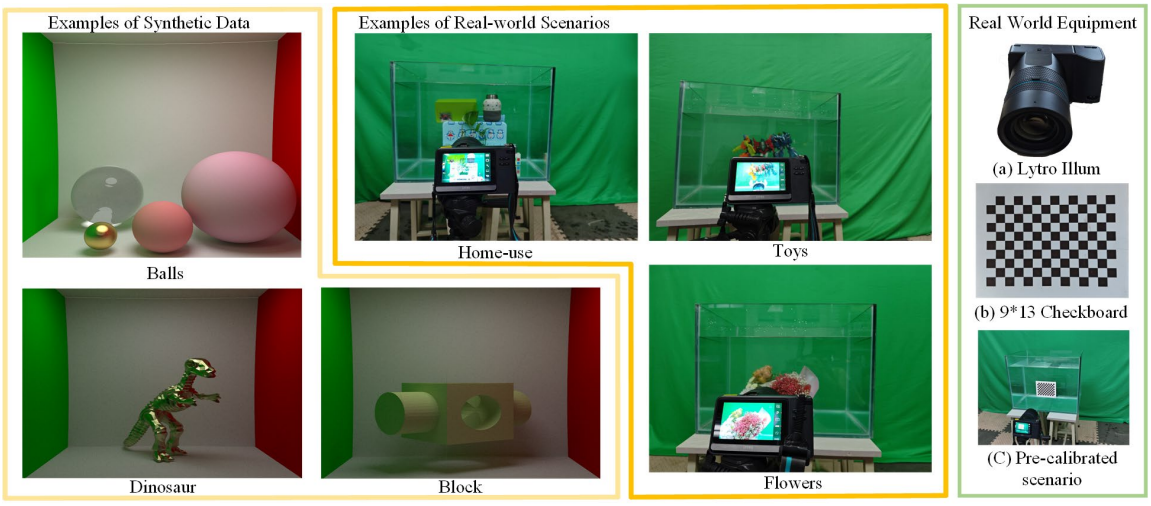

Figure 3. Experimental scenes and equipment

We used the rendering programming toolkit Taichi[6] to render balls, metal dinosaurs, wooden blocks, etc. of different materials and sizes in Cornell boxes to generate synthetic datasets of underwater scenes, as shown in Figure 3. The simulation scene uses ray tracing to simulate the refraction, reflection, diffuse reflection, and other phenomena of light, which strictly conforms to the physical behavior of light. The parameters are known at generation time and 3*3 array images are generated for each scene.

Figure 3 shows that home-use, toys, flowers, etc. are placed in the water, the imaging device is directly in front of the tank, and the view is roughly focused on the center of the tank. A checkerboard calibration pattern is posted on the surface of the water tank, and the imaging device takes the calibration pattern at a fixed position for camera parameter calibration. A light field camera is used as imaging device to capture real underwater datasets. The special sensor and lens design of the light field camera installs some microlenses between the image sensor and the main lens, each of which captures a different view of the scene and encodes the information of these views into the final image. The light field camera model in the experiment is Lytro Illum with a total of 15×15 sub-views. The reference method[24] was used to obtain the decoded data of each sub-view. As shown in Figure. 4, these 9 sub-views are selected as a 3×3 camera array in this paper.

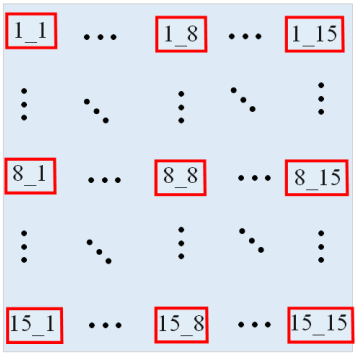

Figure 4. Light field camera 3×3 sub-view array image position

For both real and synthetic data, the central view of the array image is used as the test image in this paper, and the remaining eight side views are used for training. We compare our method with NeRF and some current related work on NeRF: MIP-Nerf[13], Tensorf[15], IBRNet[16], DS-NeRF[25], Plenoxels[26], NeRFReN[19]. To conduct a quantitative analysis of the new view synthesis results, we employ commonly utilized evaluation metrics derived from the generated new view. Namely, peak signal-to-noise ratio (PSNR), structural similarity index measure (SSIM)[27], and learned perceptual image patch similarity (LPIPS)[28], and "Average"[13], which is the geometric mean of $MSE = 10^{-\frac{PSNR}{10}}$, $\sqrt{1-SSIM}$ and LPIPS. Our framework is implemented based on Pytorch with the network optimizer Adam[29] with a learning rate of 0.0005 and the number of randomly sampled pixels during rendering is 500. The hardware conditions used are Intel Core i7-12700H CPU@4.70GHz ×20 processor and NVIDIA® GeForce RTX 3090Ti GPU (24GB RAM).

Comparative experiment

In this paper, four sets of synthetic data are rendered to verify the efficacy of the proposed method in synthesizing new views, which are tested in four synthetic underwater scenes and compared with the baseline method with the same experimental parameter settings. Among them, the dinosaur and the wood block have a resolution of 700×700 pixels, and the pig and the balls have a resolution of 800×800 pixels. Table 1 shows the quantitative comparison results, the bold font indicates the optimal metric, and the visualization results of the new view rendering related to the metric are depicted in Figure 5.

It is evident from the information presented in Table 1 that, in terms of synthetic data sets, the proposed method in this paper has the best PSNR for rendering new views compared with other methods. Compared with vanilla NeRF in the four evaluation indicators PSNR, SSIM, LPIPS, and Average, our method improves 54.49%, 34.31%, -20.72%, and -52.15% on average for each scene, and PSNR is also about 5 points higher than other baseline methods in the table on average. Among the three evaluation indicators, SSIM, LPIPS and Average, although not all of them reach the optimal values, they also perform well

Table 1. Quantitative comparison of different baseline methods on synthetic data

	Dinosaur				Block			
	PSNR↑	SSIM↑	LPIPS↓	Average↓	PSNR↑	SSIM↑	LPIPS↓	Average↓
NeRF[2]	25.606	0.676	0.579	0.097	21.888	0.578	0.626	0.138
Mip-NeRF[13]	24.978	0.670	0.502	0.097	30.086	0.724	0.501	0.064
IBRNet[16]	27.750	0.697	0.500	0.078	28.633	0.691	0.607	0.077
Tensorf[15]	26.711	0.670	0.554	0.088	27.356	0.673	0.597	0.086
DS-NeRF[25]	31.510	0.731	0.519	0.057	32.703	0.745	0.532	0.052
Plenoxels[26]	27.248	0.746	0.488	0.077	28.622	0.773	0.500	0.069
NeRFReN[19]	26.307	0.715	**0.478**	0.084	30.320	0.752	0.517	0.062
ours	**33.007**	**0.747**	0.487	**0.050**	**36.145**	**0.962**	**0.481**	**0.028**
	Pig				**Balls**			
	PSNR↑	SSIM↑	LPIPS↓	Average↓	PSNR↑	SSIM↑	LPIPS↓	Average↓
NeRF[2]	25.608	0.756	0.522	0.089	17.203	0.496	0.638	0.205
Mip-NeRF[13]	32.573	0.909	0.484	0.043	24.556	0.787	0.484	0.092
IBRNet[16]	30.243	0.876	0.496	0.055	26.417	0.824	0.539	0.080
Tensorf[15]	30.545	0.890	0.547	0.054	23.626	0.745	0.541	0.106
DS-NeRF[25]	34.282	0.911	0.507	0.038	30.124	**0.840**	0.497	0.058
Plenoxels[26]	31.761	0.903	0.469	0.046	29.432	0.838	**0.467**	0.059
NeRFReN[19]	29.914	0.899	0.495	0.054	24.846	0.839	0.494	0.091
ours	**36.357**	**0.922**	**0.467**	**0.031**	**31.346**	0.837	0.469	**0.052**

Column 1 of Figure 5 shows the real images of the central view participating in the detection, and according to the real values of the images marked by the red and green boxes in column 1, the local situation of the new view rendering results of different methods can be observed. Columns 2 to 9 show the magnified local details of the rendered new views of Ground Truth, NeRF, NeRFReN, IBRNet, Tensorf, Plenoxels, Mip-NeRF, and ours method.

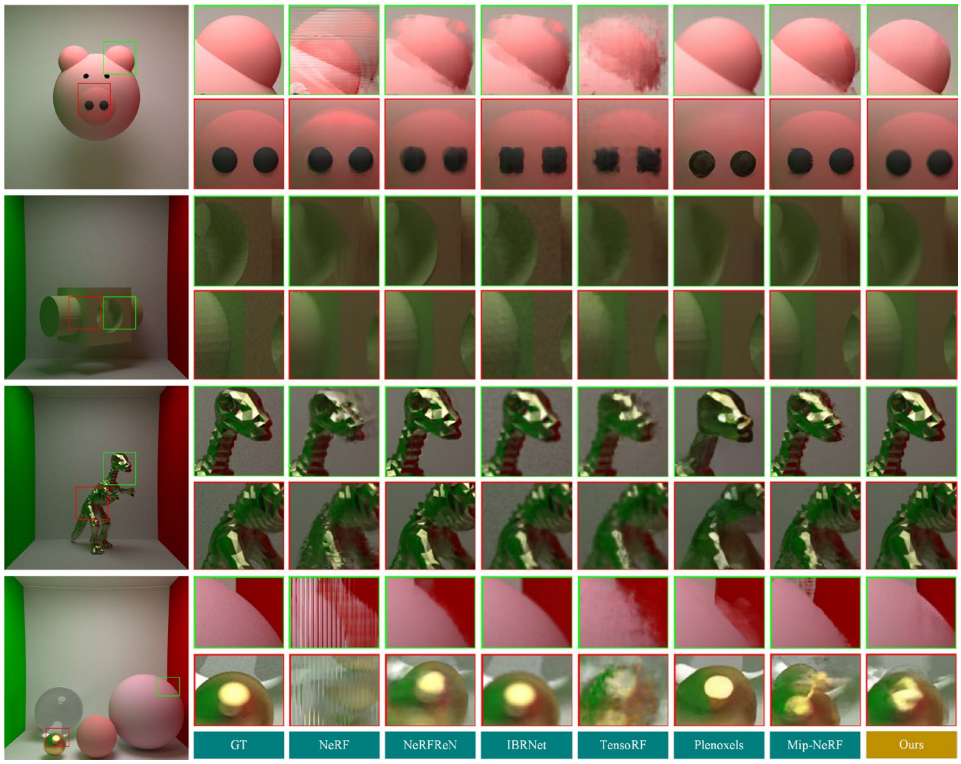

Figure 5. Qualitative Comparison of Simulation Scenarios.

Through simulation data of qualitative results, it is evident that the introduced methodology exhibits excellent view rendering ability under the condition of avoiding rendering ghost, blur, and dislocation, and the rendering details are more abundant, such as the ears of the pig, the nostrils, the texture of the wood block and the details of the dinosaur head, etc. Our method can synthesize relatively clear new views.

We continue to conduct real experiments to demonstrate the method's effectiveness further. The experiments were carried out on four real data sets composed of home-use, toys, flowers, and fruits in the water tank captured by light field camera, and compared with the baseline method with the same experimental parameter settings. The image resolution size is a fixed value of 625×434 pixels for the light field camera decoding sub-view.

Table 2. Quantitative comparison of different baseline methods on real data

	Home-use				**Toys**			
	PSNR↑	SSIM↑	LPIPS↓	Average↓	PSNR↑	SSIM↑	LPIPS↓	Average↓
NeRF[2]	23.729	0.744	0.282	0.085	28.094	0.824	0.247	0.054
Mip-NeRF[13]	21.517	0.612	0.431	0.124	24.099	0.697	0.381	0.093
IBRNet[16]	22.687	0.669	0.414	0.108	27.618	0.798	0.355	0.065
Tensorf[15]	20.907	0.576	0.463	0.134	24.006	0.693	0.403	0.096
DS-NeRF[25]	26.385	0.767	0.353	0.073	30.270	0.863	0.233	0.043
Plenoxels[26]	24.732	0.744	0.376	0.086	26.840	0.774	0.360	0.071
NeRFReN[19]	29.844	0.847	0.247	0.046	30.401	0.797	0.254	0.047
ours	**30.246**	**0.884**	**0.176**	**0.038**	**31.393**	**0.876**	**0.205**	**0.037**
	Flowers				**Fruits**			
	PSNR↑	SSIM↑	LPIPS↓	Average↓	PSNR↑	SSIM↑	LPIPS↓	Average↓
NeRF[2]	27.244	0.782	0.300	0.064	29.644	0.842	0.232	0.046
Mip-NeRF[13]	21.803	0.548	0.442	0.125	27.267	0.772	0.339	0.067
IBRNet[16]	28.988	0.794	0.271	0.054	29.473	0.863	0.177	0.039
Tensorf[15]	22.959	0.548	0.472	0.117	28.546	0.794	0.342	0.060
DS-NeRF[25]	27.407	0.746	0.361	0.069	31.669	0.861	0.290	0.042
Plenoxels[26]	24.345	0.674	0.422	0.096	-	-	-	-
NeRFReN[19]	30.252	0.799	0.259	0.048	-	-	-	-
ours	**31.256**	**0.870**	**0.228**	**0.040**	**34.020**	**0.900**	**0.143**	**0.026**

Compared with vanilla NeRF in the four evaluation indicators PSNR, SSIM, LPIPS, and Average, our method improves 54.49%, 34.31%, -20.72%, and -52.15% on average for each scene. In four real scenarios, the quantitative comparison results of the proposed method are optimal compared with other baseline methods. In real scenes, due to the scattering and absorption of light in water, the texture details of underwater objects become less, and the feature points between views are reduced. Some methods are not suitable for objects with few feature points in refraction scenes, but the proposed method still shows strong applicability in this respect.

Figure 6 shows the qualitative results of four real scene comparison experiments, where column 1 shows real images not involved in training, and columns 2 to 9 show the visualization results of new view rendering of locally magnified scenes. It becomes apparent that the rendering outcomes of alternative methods fall short of being satisfactory, for example, the English letters in the red box of home-use. Other methods will have problems such as missing details, overlapping, and missing features. Other baseline methods render fruits with unclear edges, mixed colors, and blurred textures. Our method solves these problems well.

CONCLUSIONS

We present a refractive neural rendering method in flat refractive geometry, which aims to solve the problem of novel view synthesis based on refractive planes. The distance from the ray origin to the refraction plane and the normal vector is pre-calibrated before performing neural rendering, and these pre-calibrated parameters are used to build a flat refractive geometry model. Then, the path information of the refracted light ray is obtained based on Snell's law, the neural radiation field can be trained to synthesize a new viewpoint of the scene under the refraction plane. We not only build synthetic and real datasets in the flat refractive scenario but also show excellent performance on both kinds of datasets.

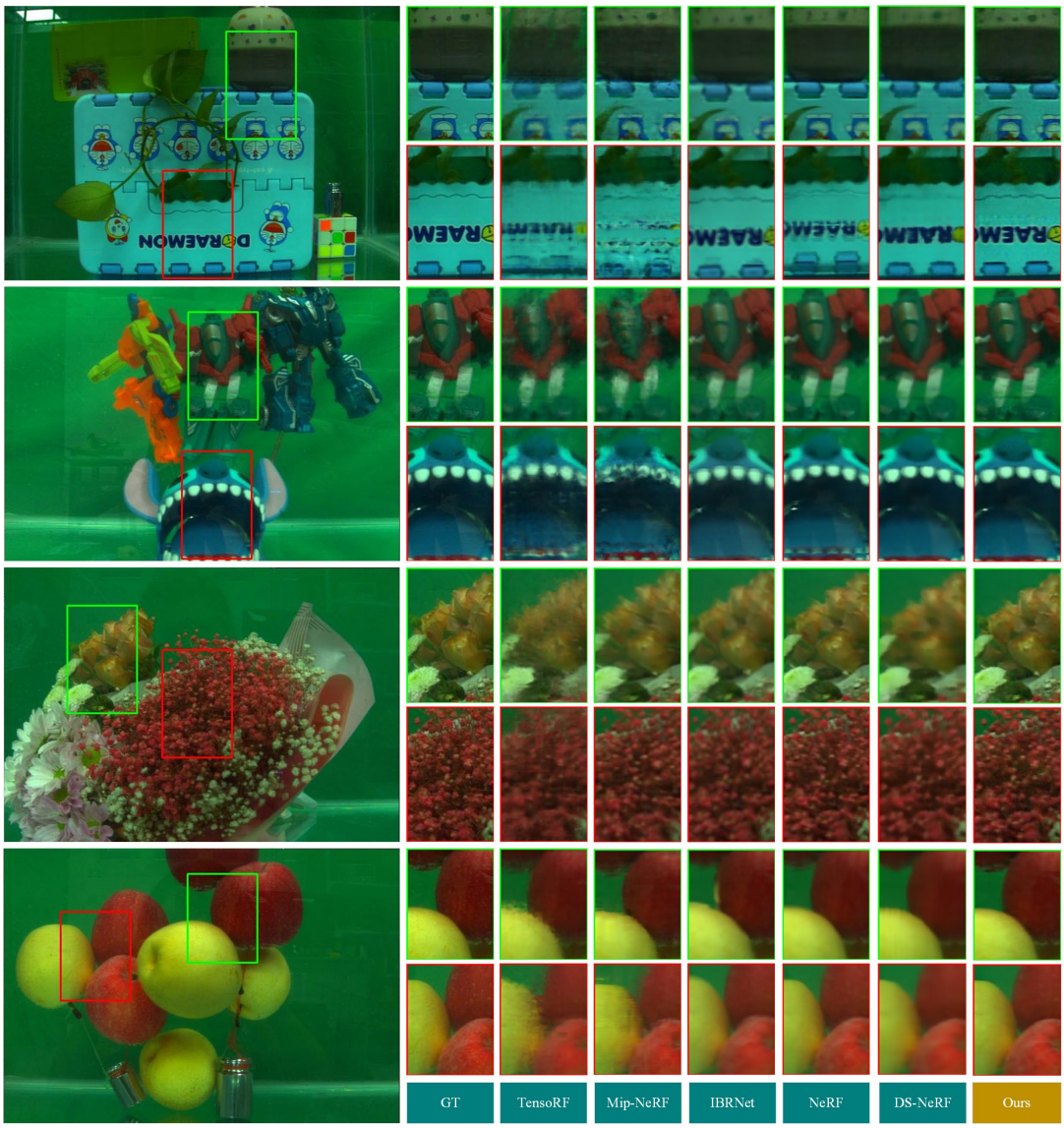

Figure 6. Qualitative Comparison of Real-world Scenarios

ACKNOWLEDGEMENTS

This work is supported in part by the Natural Science Foundation of China under Grant 62201479, in part by the Natural Science Foundation of Sichuan Province under Grant 2023NSFSC1388.

REFERENCES

[1] Huang, Yan, et al. “State-of-the-art and development trends of AUV intelligence.” Robot 42(2), 215-231(2020). 10.13973/j.cnki.robot.190392

[2] Mildenhall, Ben, et al. “Nerf: Representing scenes as neural radiance fields for view synthesis.” Communications of the ACM 65(1), 99-106(2021). 10.1007/978-3-030-58452-8_24

[3] Fujitomi T, Sakurada K, Hamaguchi R, et al. “LB-NERF: light bending neural radiance fields for transparent medium.” 2022 IEEE International Conference on Image Processing (ICIP), IEEE, 2142-2146(2022). 10.1109/ICIP46576.2022.9897642

[4] Wang Z, Yang W, et al. “NeReF: Neural Refractive Field for Fluid Surface Reconstruction and Rendering.”2023 IEEE International Conference on Computational Photography (ICCP). IEEE, 1-11(2023). 10.1109/ICCP56744.2023.10233838
[5] Zhan Y, Nobuhara S, Nishino K, et al. “Nerfrac: Neural radiance fields through refractive surface.” Proceedings of the IEEE/CVF International Conference on Computer Vision, 18402-18412(2023). 10.1109/ICCV51070.2023.01687
[6] Hu, Yuanming, et al. “Taichi: a language for high-performance computation on spatially sparse data structures.” ACM Transactions on Graphics (TOG) 38(6), 1-16(2019). 10.1145/3355089.3356506
[7] Agrawal A, Ramalingam S, Taguchi Y, et al. “A theory of multi-layer flat refractive geometry.” 2012 IEEE conference on computer vision and pattern recognition. IEEE, 3346-3353(2022). 10.1109/CVPR.2012.6248073
[8] Chen X, Yang Y H. “A closed-form solution to single underwater camera calibration using triple wavelength dispersion and its application to single camera 3D reconstruction”. IEEE Transactions on Image Processing26(9), 4553-4561(2017). 10.1109/TIP.2017.2716194
[9] Łuczyński, Tomasz, Max Pfingsthorn, and Andreas Birk. “The pinax-model for accurate and efficient refraction correction of underwater cameras in flat-pane housings.” Ocean Engineering 133, 9-22(2017). https://doi.org/10.1016/j.oceaneng.2017.01.029
[10] Kang L, Wu L, Wei Y, et al. “Two-view underwater 3D reconstruction for cameras with unknown poses under flat refractive interfaces.” Pattern Recognition 69, 251-269. https://doi.org/10.1016/j.patcog.2017.04.006
[11] Pedersen M, Hein Bengtson S, Gade R, et al. “Camera calibration for underwater 3D reconstruction based on ray tracing using Snell's law.” Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, 1410-1417(2018). 10.1109/CVPRW.2018.00190
[12] Zhang Xiaoqiang, Zhong Liangtao, Leng Qiqi, et al. “Underwater light field camera calibration based on multi-layer flat refractive model and multi-projection-center model.” Acta Optica Sinica 42(12), 1215001 (2022). 10.3788/aos202242.1215001
[13] Barron J T, Mildenhall B, Tancik M, et al. “Mip-nerf: A multiscale representation for anti-aliasing neural radiance fields.” Proceedings of the IEEE/CVF international conference on computer vision, 5855-5864(2021). 10.1109/ICCV48922.2021.00580
[14] Barron J T, Mildenhall B, Verbin D, et al. “Mip-nerf 360: Unbounded anti-aliased neural radiance fields.” Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 5470-5479(2022). 10.1109/CVPR52688.2022.00539
[15] Chen A, Xu Z, Geiger A, et al. “Tensorf: Tensorial radiance fields.” European conference on computer vision, Cham: Springer Nature Switzerland, 333-350(2022). https://doi.org/10.1007/978-3-031-19824-3_20
[16] Wang Q, Wang Z, Genova K, et al. “Ibrnet: Learning multi-view image-based rendering.” Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 4690-4699(2021). 10.1109/CVPR46437.2021.00466
[17] Sethuraman A V, Ramanagopal M S, Skinner K A. “Waternerf: Neural radiance fields for underwater scenes.” OCEANS 2023-MTS/IEEE US Gulf Coast. IEEE, 1-7(2023). 10.23919/OCEANS52994.2023.10336972
[18] Zhang T, Johnson-Roberson M. “Beyond nerf underwater: Learning neural reflectance fields for true color correction of marine imagery.” IEEE Robotics and Automation Letters, 2023. 10.1109/LRA.2023.3307287
[19] Guo Y C, Kang D, Bao L, et al. “Nerfren: Neural radiance fields with reflections.” Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 18409-18418(2022). 10.1109/CVPR52688.2022.01786
[20] Ma L, Li X, Liao J, et al. “Deblur-nerf: Neural radiance fields from blurry images.” Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 12861-12870(2022). 10.1109/CVPR52688.2022.01252
[21] Müller T, Evans A, Schied C, et al. “Instant neural graphics primitives with a multiresolution hash encoding.” ACM transactions on graphics (TOG), 41(4), 1-15(2022). 10.1145/3528223.3530127
[22] Mildenhall B, Srinivasan P.P, Ortiz-Cayon R, et al. “Local light field fusion: Practical view synthesis with prescriptive sampling guidelines.” ACM Transactions on Graphics (ToG), 38(4), 1-14(2019). 10.1145/3306346.3322980
[23] Zhang Z. “Flexible camera calibration by viewing a plane from unknown orientations.” Proceedings of the seventh ieee international conference on computer vision. Ieee, 1, 666-673(1999). 10.1109/ICCV.1999.791289
[24] Dansereau D G, Pizarro O, Williams S B. “Decoding, calibration and rectification for lenselet-based plenoptic cameras.” Proceedings of the IEEE conference on computer vision and pattern recognition, 1027-1034(2013). 10.1109/CVPR.2013.137

[25] Deng K, Liu A, Zhu J Y, et al. “Depth-supervised nerf: Fewer views and faster training for free.” Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 12882-12891(2022). 10.1109/CVPR52688.2022.01254
[26] Fridovich-Keil S, Yu A, Tancik M, et al. “Plenoxels: Radiance fields without neural networks.” Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 5501-5510(2022). 10.1109/CVPR52688.2022.00542
[27] Wang Z, Bovik A C, Sheikh H R, et al. “Image quality assessment: from error visibility to structural similarity.” IEEE transactions on image processing, 13(4), 600-612(2004). 10.1109/TIP.2003.819861
[28] Zhang R, Isola P, Efros A.A, et al. “The unreasonable effectiveness of deep features as a perceptual metric” Proceedings of the IEEE conference on computer vision and pattern recognition, 586-595(2018). 10.1109/CVPR.2018.00068
[29] Kingma D P, Ba J. “Adam: A method for stochastic optimization.” arXiv preprint arXiv:1412.6980, 2014. https://doi.org/10.48550/arXiv.1412.6980