Computers, Materials & Continua

Tech Science Press

DOI: 10.32604/cmc.2024.053115

ARTICLE

RWNeRF: Robust Watermarking Scheme for Neural Radiance Fields Based on Invertible Neural Networks

Wenquan Sun[1,2], Jia Liu[1,2,*], Weina Dong[1,2], Lifeng Chen[1,2] and Fuqiang Di[1,2]

[1]Department of Cryptographic Engineering, Engineering University of PAP, Xi'an, 710086, China

[2]Department of Cryptographic Engineering, Key Laboratory of PAP for Cryptology and Information Security, Xi'an, 710086, China

*Corresponding Author: Jia Liu. Email: liujia1022@gmail.com

Received: 24 April 2024 Accepted: 11 July 2024 Published: 12 September 2024

ABSTRACT

As neural radiance fields continue to advance in 3D content representation, the copyright issues surrounding 3D models oriented towards implicit representation become increasingly pressing. In response to this challenge, this paper treats the embedding and extraction of neural radiance field watermarks as inverse problems of image transformations and proposes a scheme for protecting neural radiance field copyrights using invertible neural network watermarking. Leveraging 2D image watermarking technology for 3D scene protection, the scheme embeds watermarks within the training images of neural radiance fields through the forward process in invertible neural networks and extracts them from images rendered by neural radiance fields through the reverse process, thereby ensuring copyright protection for both the neural radiance fields and associated 3D scenes. However, challenges such as information loss during rendering processes and deliberate tampering necessitate the design of an image quality enhancement module to increase the scheme's robustness. This module restores distorted images through neural network processing before watermark extraction. Additionally, embedding watermarks in each training image enables watermark information extraction from multiple viewpoints. Our proposed watermarking method achieves a PSNR (Peak Signal-to-Noise Ratio) value exceeding 37 dB for images containing watermarks and 22 dB for recovered watermarked images, as evaluated on the Lego, Hotdog, and Chair datasets, respectively. These results demonstrate the efficacy of our scheme in enhancing copyright protection.

KEYWORDS

Neural radiance fields; 3D scene; robust; watermarking; invertible neural networks

1 Introduction

Implicit Neural Representation (INR), also known as coordinate-based representation, offers a method of parameterizing various signals. Unlike traditional discrete signal representations, implicit neural representations characterize a signal as a continuous function. Among the applications of INR, Neural Radiance Fields (NeRF) [1] stand out as a prominent example. NeRF represents a deep learning model for 3D implicit spatial modeling, utilizing neural networks to implicitly capture the color and density functions of each point within a 3D scene. Ongoing research in NeRF focuses on

enhancing the quality of 3D content representation [2–5], accelerating rendering processes [6–8], and reconstructing sparse views [9–12]. As NeRF advances in 3D content representation, the need for addressing copyright protection concerns surrounding implicitly representation-oriented 3D models of neural radiance fields has become increasingly urgent.

In the future, individuals will likely share their captured 3D content online, akin to sharing images and videos, necessitating measures to protect the copyrights of NeRF and 3D scenes shared online to prevent unauthorized reposting. StegaNeRF [13] pioneered the fusion of the neural radiance field with information hiding. Employing a two-step training approach, it embeds secret information within non-essential weight parameters of the network, ensuring that images rendered from NeRF carry watermarking information, subsequently extracted using a dedicated network. However, StegaNeRF's direct alteration of network parameters compromises NeRF's ability to represent 3D content, thus diminishing rendered image quality. Given the growing importance of retrieving information from 2D renderings of 3D models in domains like gaming, film production, and graphic design, maintaining the quality of rendered images becomes paramount.

To tackle this issue, we propose a novel approach aimed at safeguarding the copyright of the neural radiation fields through the application of a robust watermarking technique utilizing reversible neural networks. The envisioned application scenarios of this approach involve various instances. For instance, Alice captures images of a 3D scene through photography, embeds watermarks into these images, and subsequently utilizes a Neural Radiance Fields (NeRF) model for rendering the scene. She then disseminates both the NeRF model and the rendered 3D scene online for public enjoyment. However, if Bob, without Alice's authorization, appropriates the NeRF model and shares it under his name, Alice observes the NeRF model shared by Bob and utilizes it to generate a 2D image for watermark extraction, thereby confirming Alice's ownership of the copyright for the NeRF model. Bob is found to be infringing on the copyright and is required to remove the post, as depicted in Fig. 1.

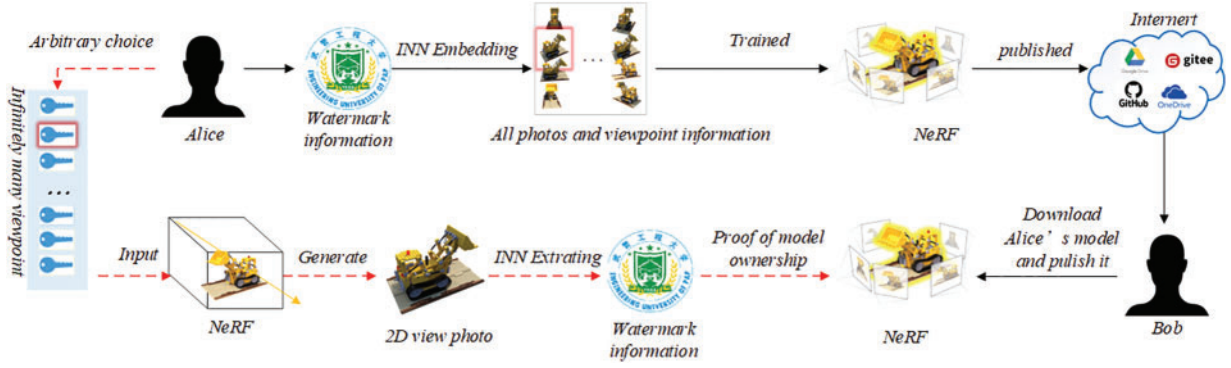

Figure 1: Application scenarios

Our primary contributions are threefold:

Firstly, compared to the latest NeRF copyright protection method, StegaNeRF, which directly alters NeRF's network parameters, this modification impacts the neural network's 3D content representation capability and the quality of its 3D module. Conversely, our approach refrains from modifying NeRF's network parameters. Leveraging image watermarking technology, we preserve the network's 3D content representation ability while concurrently achieving copyright protection for the NeRF model.

Secondly, to fortify the scheme's robustness, the copyright verification process incorporates a trained image quality enhancement network. This network performs image restoration on NeRF-rendered images, countering both the effects of NeRF rendering and deliberate tampering on image quality.

The operational procedure of our scheme entails the initial application of a forward network watermarking algorithm employing reversible neural networks. This algorithm embeds watermark information into each image within the training set used for NeRF model training. Subsequently, 3D modeling is conducted utilizing the NeRF model. Finally, for copyright verification, the embedded watermark information is extracted using the inverse process, facilitated by the extraction network. In scenarios where the 3D model is utilized without authorization, the verifier can extract the watermark information, thereby verifying the model's copyright, provided it is viewed from a perspective consistent with the training set.

2 Related Work

2.1 Traditional 3D Watermarking

Watermarking techniques for traditional 3D models are primarily categorized into two types: 3D mesh model-based watermarking algorithms [14–18] and 3D point cloud model-based watermarking algorithms. The former typically employ a multi-resolution framework to conduct wavelet decomposition or Fourier transform on the target triangular or polygonal mesh. Subsequently, watermark embedding is achieved either by modifying the topological or geometric features of the mesh model or by establishing a correlation function between the mesh vertices. On the other hand, the latter, as exemplified by the 3D point cloud model-based watermarking algorithm [19], initiates by establishing synchronization relationships between the point clouds. Then, the model is segmented into spherical rings based on radial radius, and the watermark is repetitively inserted into the vertices of each spherical ring to accomplish watermark embedding.

2.2 Neural Network Watermarking

The representation of 3D models in NeRF differs significantly from traditional approaches, as NeRF bypasses conventional geometric structures and instead directly learns and produces lifelike renderings through neural networks. Essentially, NeRF functions as a neural network for the implicit representation of 3D scenes. Consequently, traditional 3D model watermarking algorithms do not apply to watermarking neural radiance fields. The protection of copyrights for neural networks, termed neural network watermarking, has emerged as a critical research avenue in the security domain. Neural network watermarking encompasses four primary types: white-box watermarking, black-box watermarking, boxless watermarking, and vulnerable neural network watermarking. In the white-box watermarking scheme [20], the verifier can inspect the network's internals and access information such as weights to authenticate its copyright. The black-box watermarking scheme [21] suits scenarios where the verifier lacks direct access to the network's internals but can interact with it via a remote API interface. Boxless watermarking [22] primarily serves for copyright validation in generative networks, training the network to embed watermark information directly into generated images for direct copyright verification by the verifier. Vulnerable watermarking [23] diverges from the aforementioned three by detecting malicious tampering with the network's functionality, such as injecting a backdoor, based on watermark corruption.

2.3 Invertible Neural Networks Watermarking

INNs (Invertible Neural Networks) are neural networks designed using invertible transformation, and normalizing flow achieves the invertible transformation between the data distribution px and a latent distribution px [24]. Jing et al. [25] pioneered the fusion of reversible neural networks with information hiding, treating the concealment and retrieval of a secret message as complementary processes and employing the same network for both tasks. Addressing the escalating issue of artwork plagiarism, Luo et al. [26] advocated for copyright protection for high-quality artworks, integrating reversible neural networks into watermarking to bolster copyright security with heightened concealment and resilience. However, Ma et al. [27] critiqued the robustness of such schemes, attributing vulnerabilities to excessive reliance on reversibility. Consequently, a hybrid watermarking scheme intertwining reversible and non-reversible networks was devised, featuring a watermark extractor leveraging the attention mechanism across multiple channels to optimize watermark extraction efficacy and enhance scheme robustness.

3 Network Components

Our proposed watermarking method comprises four primary network structures: Invertible Block, FDTM (Frequency Domain Transform Module), NeRF (Neural Radiation Fields), and IQEM (Image Quality Enhancement Module). The invertible block facilitates watermark embedding and extraction. The frequency domain transform module preprocesses training images with watermark information. Our method safeguards the neural radiation fields. The image quality enhancement module is utilized to counteract losses in the rendering process as well as losses due to malicious attacks during propagation.

3.1 Overall Framework

We propose the utilization of an invertible neural network 2D watermarking algorithm to safeguard both the neural radiation field and the 3D scene. The algorithmic framework comprises a frequency domain transform module, invertible module, neural radiation field, and image quality enhancement block as depicted in Fig. 2. The embedding and extraction processes in invertible neural network watermarking constitute a pair of inverse operations.

$$I_W = H\left(I, M_W\right) \tag{1}$$

$$\left(I_R, R_W\right) = H^{-1}\left(QEM\left(NeRF\left(I_W\right)\right)\right) \tag{2}$$

In Eqs. (1) and (2): where H(·) represents the forward embedding watermarking process and $H^{-1}(\cdot)$ signifies the reverse extraction watermarking process. IQEM serves as the image quality enhancement module, while NeRF represents our protected target. The process begins with operating on the training image I along with the watermark information M_W using operation H(·) to yield I_W. Subsequently, I_W is fed into NeRF for rendering. The resulting image is then subjected to enhancement by IQEM. Finally, the watermark information R_w is recovered through the process $H^{-1}(\cdot)$.

In the forward embedding watermarking process, the training image I and the watermark information MW serve as inputs. Initially, they undergo the DWT (Discrete Wavelet Transform), decomposing it into high and low-frequency wavelet subbands, which are then fed into a sequence of invertible blocks. Following the final invertible block output, the IWT (Inverse Wavelet Transform) is applied to generate the watermarked image I_W along with the loss information l. All images utilized for NeRF training undergo these operations to ensure that watermarking information can be extracted

from any angle within the training set. The resulting watermarked image I_W is then employed to train the NeRF model, with specified camera position, orientation, and field of view parameters for rendering. The rendered image is produced through ray-voxel intersection sampling, color blending operation, and subsequently subjected to noise layer processing to obtain the corresponding distorted image I'.

In the reverse extraction watermarking process, the distorted image I' undergoes initial processing through IQEM to mitigate distortion effects arising from the NeRF rendering process and intentional damage modeling using the noise layer. After this, akin to the embedding process, the INN reverse process introduces a random variable Z. This variable Z is randomly sampled from an arbitrary Gaussian distribution, which matches the distribution of l. This variable Z is learned from the extraction loss during training. The auxiliary variable Z and the enhanced image I'' traverse through a frequency domain transform and a sequence of invertible blocks to yield the restored watermark R_W and the restored image I_R.

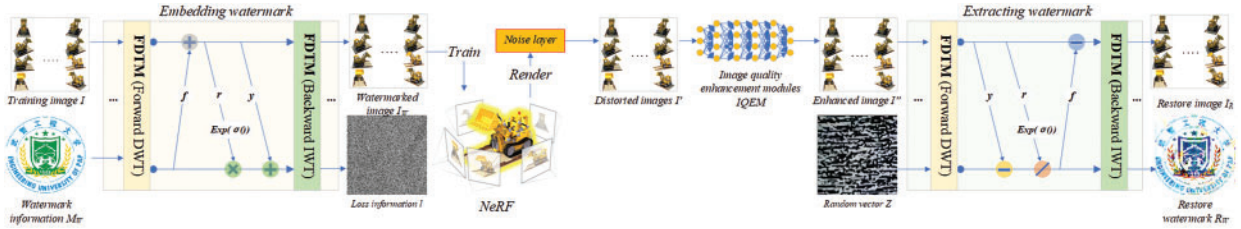

Figure 2: The overall framework of robust watermarking scheme for neural radiance fields based on invertible neural networks

3.2 Invertible Blocks

As depicted in Fig. 3, both the concealment and recovery processes entail identical sub-blocks and share network parameters, differing only in the direction of information flow. The network architecture comprises 8 invertible blocks with the same structure, outlined as follows:

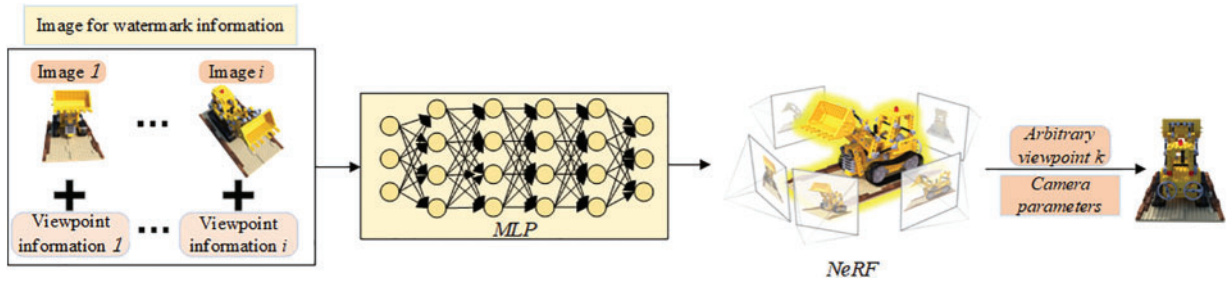

Figure 3: The detailed workflow of NeRF

For the L^{th} concealment block in the forward process, the inputs encompass I^l and $M^l{}_W$, yielding outputs I^{l+1} and $M^{l+1}{}_W$.

$$I^{l+1} = I^l + f\left(M_W^l\right) \tag{3}$$

$$M_W^{l+1} = M_W^l \otimes \exp\left(\sigma\left(r\left(I^{l+1}\right)\right)\right) + y\left(I^{l+1}\right) \tag{4}$$

In Eqs. (3) and (4): where σ denotes the activation function, LeakyReLU (Leaky Rectified Linear Unit) in this context. Functions $f(\cdot)$, $r(\cdot)$ and $y(\cdot)$ represent densely connected networks. The outputs of the final invertible block, $M^k{}_W$, and I^k, subsequently undergo transformation via IWT to yield the watermarked image I_W, and loss information l.

The L^{th} recovery block in the reverse recovery process, with inputs I''^{l+1} and Z^{l+1}, and outputs I''^{l} and Z^{l}, is presented in Eqs. (5) and (6).

$$Z^l = \left(Z^{l+1} - y\left(I''^{l+1}\right)\right) \otimes \exp\left(-\sigma\left(r\left(I''^{l+1}\right)\right)\right) \tag{5}$$

$$I''^l = I''^{l+1} - f\left(Z^l\right) \tag{6}$$

where σ denotes the activation function, LeakyReLU in this context. Functions $f(\cdot)$, $r(\cdot)$ and $y(\cdot)$ represent densely connected networks. In the reverse process, the information flow direction is reversed compared to the forward process, traversing through the $l{+}1^{th}$ layer before the l^{th} layer. Finally, following the initial invertible transformation layer, the data undergoes IWT to derive the restored image I_R and the restored watermark R_W.

3.3 Frequency Domain Transform Module

Watermarked images embedded in the pixel domain are susceptible to texture replication artifacts and color distortion [28,29]. The frequency domain and high-frequency domain are more conducive to watermark embedding compared to the pixel domain. We employ the FDTM to partition the image into low-frequency and high-frequency wavelet subbands before the invertible transform. The high-frequency subbands encapsulate image details, while the low-frequency subbands encompass overall image features, facilitating improved fusion of watermark information into the carrier image by the network. Compared to direct operations in the original image domain, wavelet transforms offer superior visual fidelity and embed watermark information in a few subbands, minimally impacting the overall image and often eluding detection. Additionally, the excellent reconstruction properties of wavelets [30] mitigate information loss and enhance watermark embedding capabilities. Before entering the invertible block, the image undergoes processing through the FDTM. Following the DWT, the feature map of size (B,C,H,W) transforms into (B,4C,H/2,W/2), where B represents the batch size, H denotes height, W signifies width, and C represents the number of channels. The DWT effectively reduces computational costs, thereby expediting the training process. After the last invertible block, the feature map (B,4C,H/2,W/2) is fed into the FDTM for IWT, thereby restoring the feature map size to (B,C,H,W) and generating the restored watermark I_W.

3.4 Neural Radiation Fields

NeRF is a neural network model designed for generating 3D scenes. The network structure comprises multiple layers of perceptrons, which are employed to encode the scene's surface as depicted in Fig. 3. In the neural radiation fields model, each pixel position of the input image can be represented as a 3D coordinate point in the scene, allowing for precise object location and rendering within the scene. In NeRF, the input spatial point is defined by a 3D coordinate position, $x = (x, y, z)$, and a direction, $d = (\theta, \phi)$, while the output spatial point is characterized by a color, $c = (r, g, b)$, and density σ at the corresponding voxel position.

$$F_\theta : \ (x\,(x, y, z), d\,(\theta, \phi)) \rightarrow (c\,(r, g, b), \sigma) \tag{7}$$

In Eq. (7): where $x = (x, y, z)$ for the position in 3D space, $d = (\theta, \varphi)$ for the line-of-sight angle, and $c = (r, g, b)$ for the color of the corresponding position, σ the density at the position of

the corresponding voxel. NeRF takes in a finite sequence of discrete images and camera parameters associated with specific viewpoints to generate a continuous static 3D scene. Moreover, it can render the scene from infinite perspectives, resulting in new viewpoint images. Body rendering, on the other hand, is a 3D-to-2D modeling process that leverages the pixel values c and the body density σ of 3D points obtained through 3D reconstruction. The final pixel values of the 2D image are derived by the weighted superposition of pixel point samples along a ray in the direction of observation.

$$C(r) = \int_{t_n}^{t_f} T(t)\,\sigma(r(t))\,c(r(t), d)\,dt, \text{where } T(t) = \exp\left(-\int_{t_n}^{t_f} \sigma(r(s))\,ds\right) \tag{8}$$

This process is illustrated in Eq. (8): where the ray, denoted as $r(t)$, is defined as $r(t) = o + td$. o represents the position of the camera's optical center, and d represents the direction of the viewing angle. Furthermore, $T(t)$ indicates the cumulative transmittance of the ray as it travels from the proximal point t_n to the distal boundary t_f. Building upon this characteristic of NeRF, this paper proposes a method for extracting watermarks from any angle in the training set by randomly selecting camera parameters. This approach aims to provide copyright protection for NeRF.

3.5 Image Quality Enhancement Module

The invertible neural network relies on the reversibility between its forward output and reverse input. However, during actual transmission, the input for reverse propagation deviates from the output of forward propagation. This discrepancy arises from two primary factors: firstly, the NeRF rendering process induces partial loss of watermark information, and secondly, human-induced noise attacks during transmission preceding the NeRF network leads to further loss of watermark information. Consequently, before the reverse process for watermark extraction, we introduce an IQEM to mitigate the effects of both NeRF rendering distortions and deliberate tampering as depicted in Fig. 4. The IQEM employs a residual convolutional codec network, the Conv block in the encoder consists of a 3×3 convolution layer, a batch norm layer, and a ReLU (Rectified Linear Unit) layer. In addition, the stride and padding of the convolution layer are both set to 1, we utilize a convolutional encoder on the left side to extract multi-level features from the distorted image I'. These extracted features are subsequently input into the right inverse convolutional decoder, along with residuals from the previous layer. Overlaying the final result completes the restoration process of the image. Integration of IQEM into the watermark extraction procedure fulfills preprocessing the rendered image I' before its input into the invertible neural network. This preprocessing step ensures that the inputs propagated backward closely resemble the watermarked image I_W, thereby enhancing the scheme's capacity to extract watermark information more comprehensively.

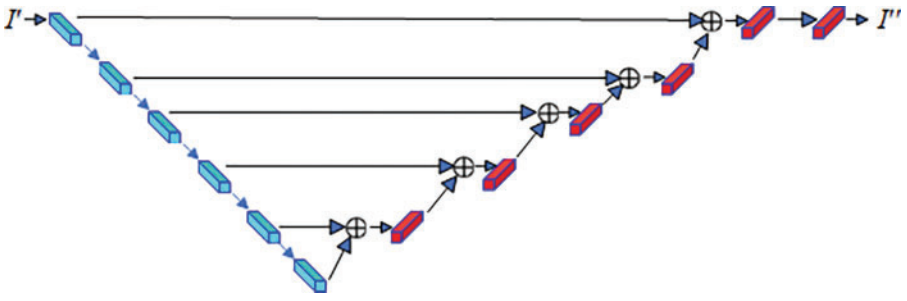

Figure 4: Image quality enhancement module architecture

4 Loss Function

The loss associated with network model training proposed in this paper consists of four main components.

4.1 Embedding Loss L_{Emb}

The purpose of the embedding loss is to ensure that the generated watermarked image I_W is indistinguishable from the training image I. The embedding loss is used in the following steps:

$$L_{Emb}(\theta) = \sum_{n=1}^{N} \ell_{Emb}\left(I_W^{(n)}, I^{(n)}\right) \tag{9}$$

In Eq. (9): where N represents the number of training samples, and ℓ_{Emb} calculates the difference between the watermarked image I_W and the training image I, we use the L_2 paradigm.

4.2 Low-Frequency Wavelet Loss $L_{low\text{-}f}$

Literature [31] verified that watermark information embedded in high-frequency components is less detectable than watermark information embedded in low-frequency components. To ensure higher visual fidelity and minimize the impact on the image as a whole due to the embedding of the watermarking information, so that the watermarking information is embedded in the high-frequency components of the image as much as possible, we employ a loss constraint on the low-frequency subbands of the training image I and the watermarked image I_W.

$$L_{low-f}(\theta) = \sum_{n=1}^{N} \ell_f\left(H\left(I^{(n)}\right)_{ll}, H\left(I_W^{(n)}\right)_{ll}\right) \tag{10}$$

In Eq. (10): where N represents the number of training samples, ℓ_f calculates the low-frequency difference between the training image I and the watermarked image I_W, and $H(\cdot)_{ll}$ represents the low-frequency subband operation of the extracted image.

4.3 Extraction Loss L_{Ext}

To ensure the consistency between the restored watermark R_W and the watermark information M_W. The difference between the restored watermark R_W and the watermark information M_W is minimized to improve the watermark extraction accuracy of the model.

$$L_{Ext}(\theta) = \sum_{n=1}^{N} E_{z\sim p(z)}\left[\ell_{Ext}\left(R_W^{(n)}, M_W^{(n)}\right)\right] \tag{11}$$

In Eq. (11): where N represents the number of training samples, and ℓ_{Ext} computes the difference between the restored watermark R_W and the watermark information M_W. The process of sampling the random vector z is random.

The total loss function of the invertible neural network is a weighted sum of the embedding loss, the low-frequency wavelet loss, and the extraction loss.

$$L_{total}(\theta) = \lambda_1 L_{Emb} + \lambda_2 L_{low-f} + \lambda_3 L_{Ext} \tag{12}$$

In the training process, λ_1, λ_2, λ_3 is a hyperparameter for making a trade-off between the L_{Emb}, L_{low-f} and L_{Ext} loss component, λ_2 is first set to 0, the network model is directly pre-trained without

considering the effect of L_{low-f} on the network, so that the network model first obtains the basic embedding-extraction ability. Then the L_{low-f} constraints are gradually added to further optimize the network model to embed the watermark information in the high-frequency components of the training image, to minimize the impact of the embedding of the watermark information on the image as a whole.

4.4 Loss of Image Quality Enhancement Module MSE

To address embedded watermark information degradation stemming from both the rendering process and simulated malicious attacks introduced by the noise layer, we constrain the loss of the IQEM using MSE (Mean Squared Error). Notably, the IQEM operates independently of the invertible neural network's training process. Its design aims to ensure that the enhanced image I'', derived from the distorted image I' via the IQEM, closely resembles the watermarked image I_W generated by the invertible neural network. This similarity is crucial for resisting watermark information corruption and loss attributable to rendering processes and noise layer attacks. The relationship is represented as:

$$MSE = \frac{1}{n}\sum_{i=1}^{n}\left(I_i^{'}, I_{Wi}\right) \tag{13}$$

In Eq. (13): where n represents the number of training samples. I'_i is the i^{th} distorted image and I_{Wi} is the i^{th} watermarked image.

5 Experimental Results

5.1 Setting

The network model employed operates on the PyTorch platform with CUDA version 11.6 and an Nvidia GeForce RTX 2070 GPU. Training for 3D scene generation utilizes the source code from Read-NeRF [1]. The architecture of the invertible neural network is derived from HiNet [25]. Training the network model involves using an Adam optimizer with hyperparameters $\lambda_1 = 5$, $\lambda_2 = 0.5$, and $\lambda_3 = 1$, a learning rate of $1\times10^{-4.5}$, and a batch size of 2. The entire network comprises 8 invertible blocks, each incorporating DenseNet blocks with 7 layers of convolutional blocks as $f(\cdot)$, $r(\cdot)$, and $y(\cdot)$ functions for coding and decoding, respectively.

5.2 Datasets

NeRF is trained using datasets such as Lego, Hotdog, Chair, etc., which use invertible neural networks to embed watermarks only on the dataset used to train NeRF. Given the diversity, high resolution, and authenticity of the DIV2K (Diverse 2K) dataset [32], we utilize it extensively. The DIV2K training dataset, consisting of 800 images at a resolution of 1024×1024, serves as the principal dataset for training the reversible neural network model. Subsequently, the network model's validation is performed using the DIV2K validation dataset, which comprises 100 images of the same resolution. Furthermore, the effectiveness of the network model is assessed using the DIV2K test dataset, which also consists of 100 images at a resolution of 1024×1024.

5.3 Performance Measurements

We use four evaluation metrics: PSNR (Peak Signal Noise Ratio), SSIM (Structural Similarity), RMSE (Root Mean Square Error), and MAE (Mean Absolute Error), to measure the watermark embedding and extraction capabilities of the network model.

PSNR is commonly used to evaluate the quality of image reconstruction and is defined by the Mean Square Error (MSE) between two images of size $W \times H$, X, and Y. The formula for PSNR is given by:

$$MSE = \frac{1}{W \times H} \sum_{i=1}^{W} \sum_{j=1}^{H} \left[X_{i,j} - Y_{i,j} \right]^2 \tag{14}$$

$$PSNR = 10 \times \log_{10} \frac{MAX^2}{MSE} \tag{15}$$

In Eqs. (14) and (15): $X_{i,j}$ and $Y_{i,j}$ refer to the pixel values of image X and Y at position (i,j), respectively. MAX represents the maximum pixel value of an image point, and a higher PSNR value indicates less distortion.

SSIM is another image quality evaluation metric that measures image similarity in terms of brightness, contrast, and structure. It is defined by:

$$\begin{aligned} l(x,y) &= \frac{2\mu_x\mu_y + C_1}{\mu_x^2 + \mu_y^2 + C_1} \\ \mathrm{c}(x,y) &= \frac{2\sigma_x\sigma_y + C_2}{\sigma_x^2 + \sigma_y^2 + \mathrm{C}_2} \\ s(x,y) &= \frac{\sigma_{xy} + C_3}{\sigma_x\sigma_y + C_3} \end{aligned} \tag{16}$$

In Eq. (16): μ_x and σ_x are the mean and variance of image X, μ_y and σ_y are the mean and variance of image Y, and σ_{xy} is the covariance of X and Y. Constants C_1, C_2, and C_3 are used, with $C_1 = (K_1 * L)\text{^}2$, $C_2 = (K_2 * L)\text{^}2$, and $C_3 = C_2/2$. In general, $K_1 = 0.01$, $K_2 = 0.03$, and $L = 255$.

$$SSIM(X, Y) = l(x,y) \cdot c(x,y) \cdot s(x,y) \tag{17}$$

SSIM values range from 0 to 1, where a higher value indicates less image distortion.

RMSE indicates the sample standard deviation of the difference between predicted and observed values (called residuals). It is equivalent to the L_2 paradigm and is more sensitive to outliers in the data.

$$RMSE = \sqrt{MSE} \tag{18}$$

MAE represents the mean of the absolute errors between predicted and observed values and is equivalent to the L_1 paradigm.

$$MAE = \frac{1}{W \times H} \sum_{i=1}^{W} \sum_{j=1}^{H} \left| X_{i,j} - Y_{i,j} \right| \tag{19}$$

5.4 Results and Analysis

To showcase the advantages and feasibility of this approach, we employ three widely recognized metrics to assess its efficacy: the invisibility of embedded watermark information, the accuracy of extracted watermark data, and the scheme's robustness against various attacks.

5.4.1 Imperceptibility

The invertible network watermarking scheme (RWNeRF) proposed achieves blind watermarking, aiming to minimize the distortion between the training image I and the watermarked image I_W. The evaluation of the method's imperceptibility utilizes four metrics: PSNR, SSIM, MAE, and RMSE. Comparisons were made between 100 training images I and their corresponding watermarked images I_W, as seen in Table 1. The experimental data demonstrate that the RWNeRF scheme successfully achieves blind watermarking.

Table 1: Comparison of imperceptibility indicators

Metrics	Datasets		
	Training image I/Watermark image I_W		
	Lego	Hotdog	Chair
PSNR	38.226542	37.379475	37.890828
SSIM	0.943185	0.918761	0.936977
MAE	3.168850	3.962571	3.114965
RMSE	5.732577	6.188527	5.956620

Meanwhile, as depicted in Fig. 5, the watermark is embedded into images from three datasets: Lego, Hotdog, and Chair, utilizing the invertible neural network robust watermarking scheme. Upon comparing the training image I with the watermarked image I_W, it is unfeasible to discern the presence of watermark information within the training image based on visual cues. Experimental findings corroborate the imperceptibility of the watermark embedded via our method, accomplishing the goal of blind watermarking.

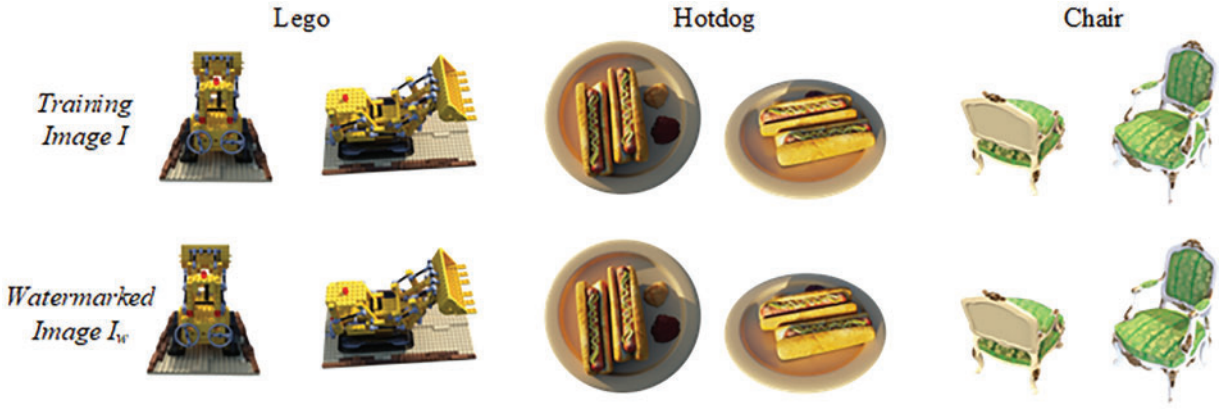

Figure 5: Visual effect obtained by applying RWNeRF for watermarking information embedding. The elements depicted in the figure included the Training image I, and the Watermarked image I_W

5.4.2 Accuracy

The RWNeRF framework embeds watermark information M_W into three datasets—Lego, Hotdog, and Chair—utilizing a forward invertible neural network. Subsequently, NeRF is trained using

the watermarked image I_W. The 3D scene is rendered with input viewpoint information, yielding the rendered image. This image undergoes processing through a noise layer to simulate malicious attacks, resulting in the distorted image I'. It is then enhanced using an image quality enhancement module to obtain the enhanced image I''. Finally, the restored watermark R_W is extracted via the reverse invertible neural network. The accuracy of the restored watermark R_W is evaluated against the watermark information M_W using four metrics as depicted in Fig. 6.

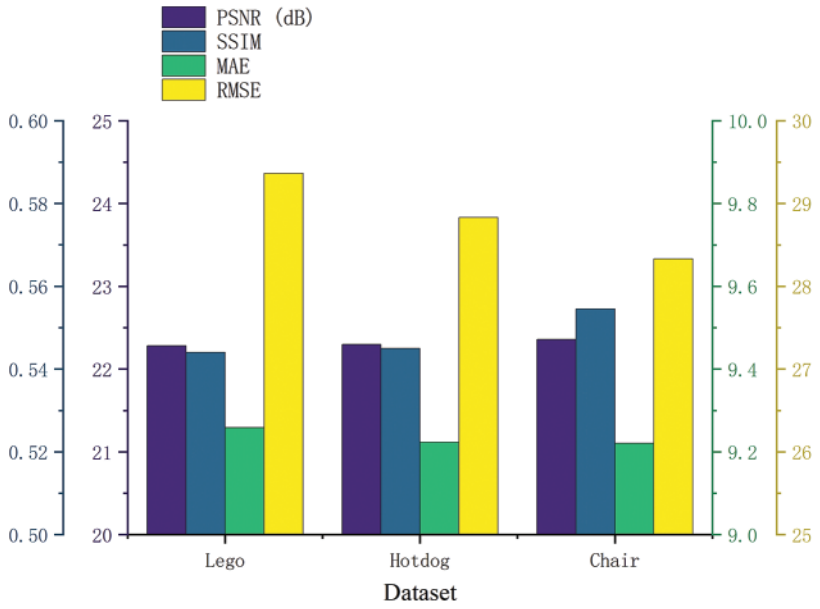

Figure 6: The average values of each metric across 100 images: PSNR exceeds 22 dB, SSIM approximates 0.55, MAE is around 9.2, and RMSE is approximately 29

RWNeRF conducts watermark extraction on original training image angles and non-original training image angles in three datasets: Lego, Hotdog, and Chair. Two parameters, Θ and Φ, influence the image angle. In this study, Θ is adjusted while Φ is fixed to control the angle change. Using the original angles $\Theta = 30$, $\Theta = 45$, and $\Theta = 60$, a view angle offset of $+1$ is applied to verify the extraction of watermark information when the selected angle differs from the original training angle. The experimental results depicted in Fig. 7 indicate that watermark information can be extracted when the selected angle matches the original training image angle. However, when the selected angle deviates from the original training image angle, i.e., other angles, the extraction of watermark information cannot be accurately achieved.

We have also explored alternative image watermarking methods for NeRF to enhance copyright protection. These methods encountered challenges such as loss of watermark information during the rendering process and potential malicious tampering during network delivery, rendering them incapable of successfully extracting watermark information, as evidenced in Table 2. Experimental results illustrate their inefficacy in this context, attributable to their development for conventional settings where 2D images serve as the ultimate visual form. In contrast, our approach addresses the novel INR framework as the foundational representation, where 2D images merely represent the final output of NeRF rendering.

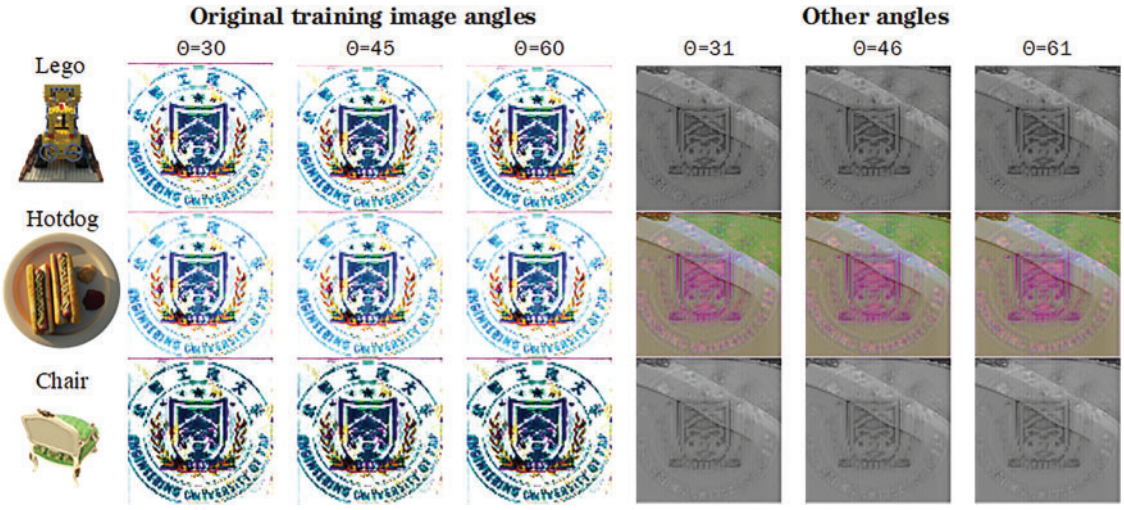

Figure 7: Comparison of the visualization of watermark extraction from different angles

Table 2: Quantitative comparison of Restore watermark R_W quality by the four methods

Dataset	Watermark information M_W/Restore watermark R_W					
	Lego		Hotdog		Chair	
Metric	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
Method						
RWNeRF	22.28	0.54	22.23	0.54	22.36	0.56
Pallaw et al. [33]	9.96	0.16	9.97	0.18	8.35	0.15
DCT-DWT [34]	7.98	0.18	9.96	0.20	7.67	0.16
Hu et al. [35]	8.81	0.15	8.74	0.14	8.99	0.17

5.4.3 Robustness

Given the unpredictable nature of distortion experienced by 3D models in practical scenarios, we employ four conventional noise attacks—Poisson, Gaussian, Pepper, and Speckle—to disrupt the rendered image, thereby simulating malicious attacks. These distortions are utilized to train our IQEM, enhancing the robustness of the proposed scheme, as depicted in Fig. 8. The rendered image undergoes disruption from the four types of noise, respectively. However, the enhanced image *I''* processed by the IQEM remains visually indistinguishable from the watermarked image I_W and cannot be identified as having suffered damage, verifying the robustness of the scheme.

Moreover, the viewpoint images generated by NeRF during the rendering process lead to the loss of original watermark information. To address this challenge, we propose an Image Quality Enhancement Module (IQEM) in contrast to conventional methods. This module is trained to mitigate various types of noise and to compensate for the loss of watermark information inherent in the rendering process. While existing methods exhibit some degree of robustness, their failure to integrate the IQEM results in an inability to compensate for watermark loss during rendering, and unable to extract watermark information successfully, as depicted in Fig. 9.

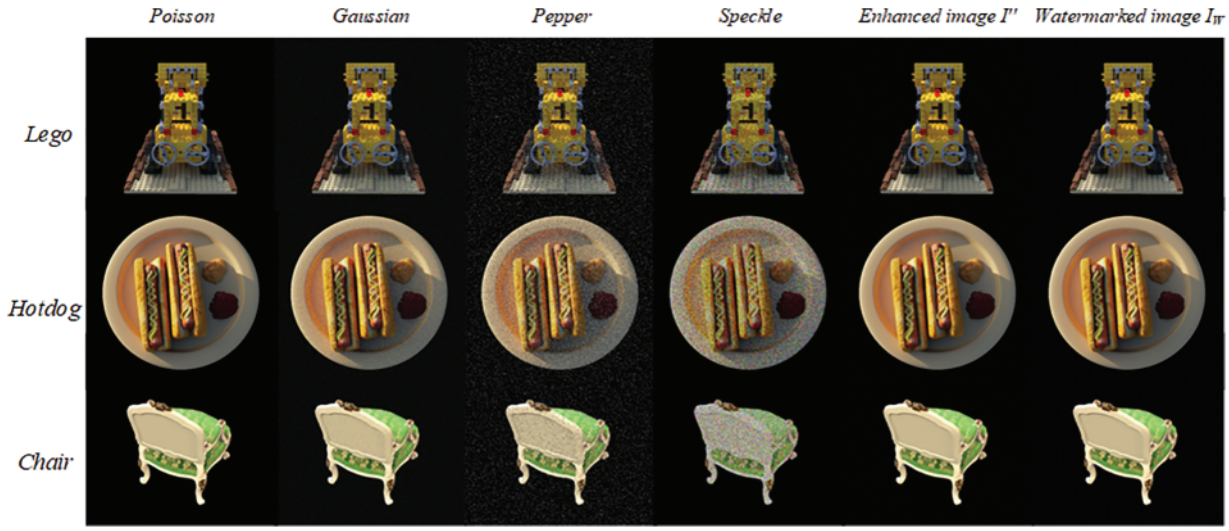

Figure 8: Visual comparison of Watermarked image I_W, Enhanced image I'' and the images after undergoing four separate noise attacks

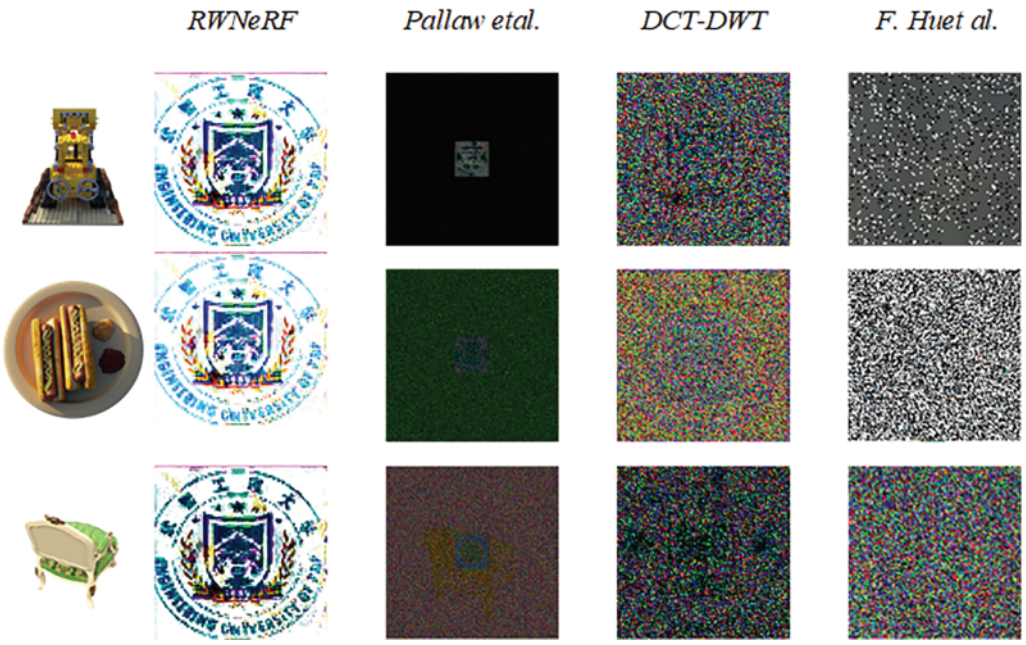

Figure 9: Comparison of the visualization of watermark extraction from different methods

5.4.4 Comparison of Rendering Image Quality

RWNeRF employs an invertible neural network watermarking approach to safeguard NeRF by embedding watermarks into the 2D images utilized for NeRF training via a forward network. Watermarks are then extracted from the rendered images using an inverse network, thus confirming NeRF's copyright. Given that direct modification of the MLP structure by StegaNeRF would impact the rendering capability of the network structure itself, RWNeRF's approach refrains from altering the

network structure directly. Consequently, it preserves NeRF's rendering capability without affecting the network structure, achieving copyright protection through an indirect method. Moreover, owing to the superior performance of reversible neural network watermarking, RWNeRF has less impact on the original training image after watermark embedding compared to traditional watermarking algorithms like LSB and other 2D watermarking algorithms based on deep learning. Through the implementation of training for the same epoch (50000), the subjective visual assessment reveals that the quality of images rendered by RWNeRF surpasses those rendered by Pallaw et al., DCT-DWT, Hu et al., and StegaNeRF, as depicted in Fig. 10.

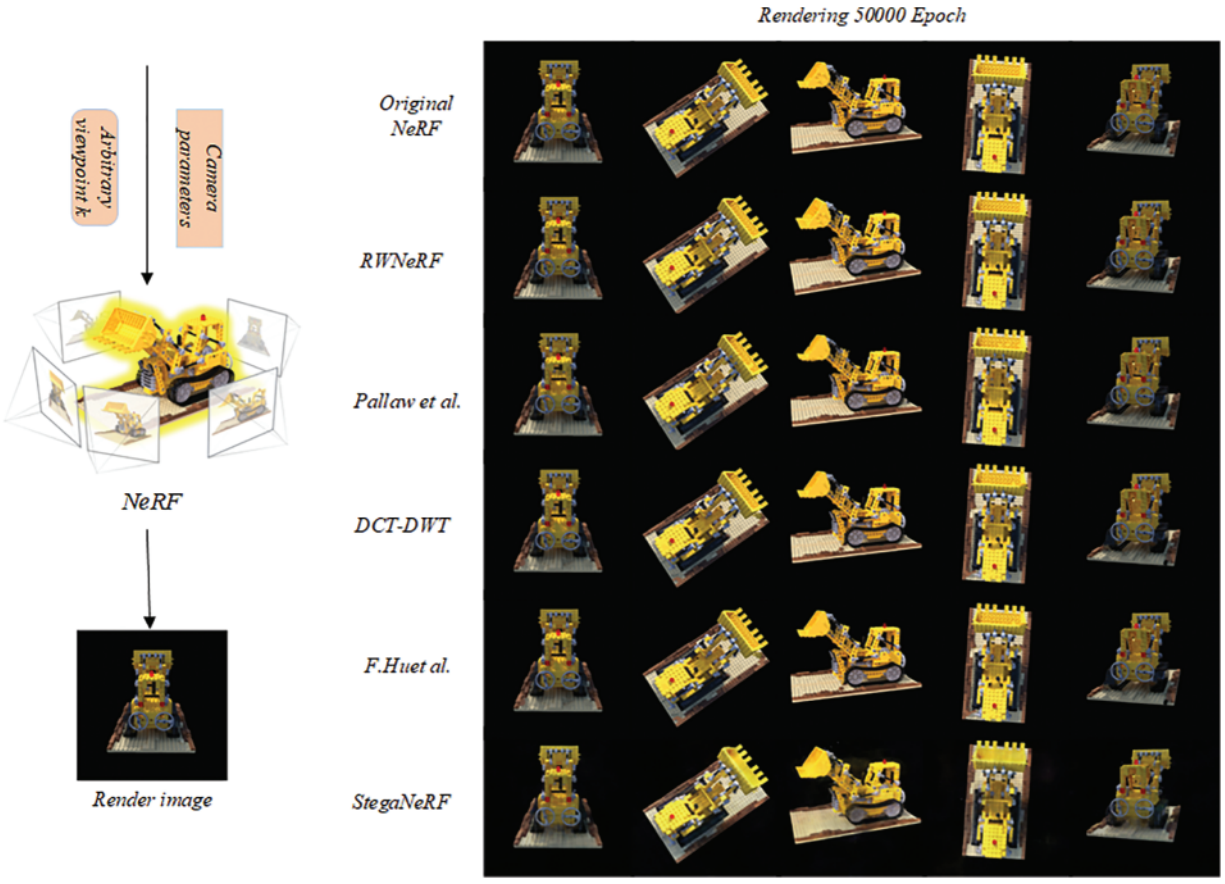

Figure 10: Visual effect obtained by applying the five methods for render image

The 13-angle images rendered by the five methods are compared with the original training images at corresponding angles, and the quantitative results of the four evaluation metrics are presented in Table 3. The standard NeRF in the first row represents the quality of the image rendered by the unembed watermark NeRF, serving as a benchmark for image quality by the five methods. The images produced by RWNeRF closely match the standard NeRF in all four evaluation metrics, demonstrating that RWNeRF can achieve copyright protection without compromising NeRF's rendering capabilities.

Table 3: Quantitative comparison of rendered image quality by the five methods

Datasets	Metrics			
	PSNR (dB)↑	SSIM↑	MAE↓	RMSE↓
Standard NeRF	32.88	0.97	3.02	7.48
RWNeRF	31.13	0.96	3.51	8.13
Pallaw et al.	29.97	0.94	5.66	10.86
DCT-DWT	30.22	0.94	4.10	9.75
Hu et al.	30.28	0.95	4.21	9.81
StegaNeRF	28.36	0.91	7.03	12.37

5.5 Ablation Study

Traditional deep learning image robust watermarking techniques such as HiNet [30] and ISN [32] are not directly applicable to our task, as they prioritize reversibility and do not consider the rendering process and simulated noise layer mimicking malicious attacks, leading to corruption of embedded watermark information in training images. Therefore, RWNeRF incorporates IQEM before watermark extraction to mitigate the effects. With the inclusion of the IQEM structure, the Peak Signal-to-Noise Ratio (PSNR) of Watermark information M_W and Restored watermark R_W improves from 5.31 to 22.23 dB, as demonstrated in Table 4. The experimental results underscore the significant value of IQEM in successfully extracting watermark information.

Table 4: Quantitative comparison of rendered image quality by the four methods

IQEM	*FDTM*	$L_{low\text{-}f}$	Comparison of watermark information M_W of Restored watermark R_W (PSNR)
×	√	√	5.31 dB
√	×	√	12.44 dB
√	√	×	19.88 dB
√	√	√	22.23 dB

6 Conclusions

In this paper, we introduce a novel approach for protecting the neural radiance field through Invertible Neural Network Robust Watermarking (RWNeRF), aiming to safeguard the copyright of NeRF. RWNeRF utilizes an invertible neural network to embed and extract watermarks on 2D images, treating the watermark embedding and extraction processes as forward and inverse operations of the reversible network. Additionally, an image quality enhancement module is integrated into the intermediate phase to compensate for the loss of watermark information resulting from the NeRF rendering process and simulated noise layer mimicking malicious attacks, thereby safeguarding the 3D model represented by the neural radiance field. Experimental findings demonstrate RWNeRF's capability in watermark embedding and extraction; however, further enhancement is needed to optimize watermark extraction quality.

Acknowledgement: In the process of designing this thesis, I am grateful to my school for providing me with the opportunity to learn. Throughout my studies, Liu Jia has provided meticulous guidance, from selecting the topic and structuring the thesis framework to detailed revisions, offering invaluable advice and suggestions. With a rigorous and pragmatic approach, Liu Jia's dedication, conscientiousness, diligence, and innovative spirit have profoundly influenced me. His profound knowledge, broad perspective, and sharp insights have deeply inspired me. This thesis was completed under his careful guidance and unwavering support. I would like to sincerely thank all the teachers who took time out of their busy schedules to review this paper.

Funding Statement: This study is supported by the National Natural Science Foundation of China, with Fund Numbers 62272478, 62102451, the National Defense Science and Technology Independent Research Project (Intelligent Information Hiding Technology and Its Applications in a Certain Field) and Science and Technology Innovation Team Innovative Research Project Research on Key Technologies for Intelligent Information Hiding" with Fund Number ZZKY20222102.

Author Contributions: The authors confirm contribution to the paper as follows: study conception and design: Jia Liu, Wenquan Sun; data collection: Lifeng Chen; analysis and interpretation of results: Weina Dong; draft manuscript preparation: Wenquan Sun; and Fuqiang Di was responsible for the finalization of the article. All authors reviewed the results and approved the final version of the manuscript.

Availability of Data and Materials: The archived version of the code described in this manuscript can be freely accessed through GitHub (https://github.com/twinlj77/IW4NeRF, accessed on 21 January 2024).

Ethics Approval: Not applicable.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi and R. Ng, "NeRF: Representing scenes as neural radiance fields for view synthesis," *Assoc. Comput. Mach.*, vol. 65, no. 1, pp. 99–106, 2021. doi: 10.1145/3503250.

[2] P. Wang, L. Liu, Y. Liu, C. Theobalt, T. Komura and W. Wang, "NeuS: Learning neural implicit surfaces by volume rendering for multi-view reconstruction," in *Proc. 35th Int. Conf. Neural Inf. Process. Syst.*, Red Hook, NY, USA, 2021, pp. 27171–27183.

[3] X. Zhang, P. P. Srinivasan, B. Deng, P. Debevec, W. T. Freeman and J. T. Barron, "NeRFactor: Neural factorization of shape and reflectance under an unknown illumination," *Assoc. Comput. Mach.*, vol. 40, no. 6, pp. 237–255, 2021. doi: 10.1145/3478513.3480496.

[4] J. T. Barron, B. Mildenhall, M. Tancik, P. Hedman, R. M. Brualla and P. P. Srinivasan, "Mip-NeRF: A multiscale representation for anti-aliasing neural radiance fields," in *2021 IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Montreal, QC, Canada, 2021, pp. 5835–5844.

[5] M. Tancik *et al.*, "Block-NeRF: Scalable large scene neural view synthesis," in *2022 IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, New Orleans, LA, USA, 2022, pp. 8238–8248.

[6] K. Schwarz, A. Sauer, M. Niemeyer, Y. Liao, and A. Geiger, "VoxGRAF: Fast 3D-Aware image synthesis with sparse voxel grids," *Adv. Neural Inform. Process. Syst.*, vol. 35, pp. 33999–34011, 2022.

[7] A. Yu, S. Fridovich-keil, M. Tancik, Q. Chen, B. Recht and A. Kazanawa, "Plenoxels: Radiance fields without neural networks," in *2022 IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, New Orleans, LA, USA, 2021, pp. 5491–5500.
[8] T. Muller, A. Evans, C. Schied, and A. Keller, "Instant neural graphics primitives with a multiresolution hash encoding," *ACM Trans. Graph.*, vol. 41, no. 4, pp. 1–15, 2022. doi: 10.1145/3528223.3530127.
[9] L. Wang *et al.*, "Fourier plenoctrees for dynamic radiance field rendering in real-time," in *2022 IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, New Orleans, LA, USA, 2022, pp. 13514–13524.
[10] C. Sun, M. Sun, and H. T. Chen, "Direct voxel grid optimization: Super-fast convergence for radiance fields reconstruction," in *2022 IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, New Orleans, LA, USA, 2022, pp. 5449–5459.
[11] D. Xu, P. Wang, Y. Jiang, Z. Fan, and Z. Wang, "Signal processing for implicit neural representations," *Adv. Neural Inf. Process. Syst.*, vol. 35, pp. 13404–13418, 2022.
[12] D. Chen, Y. Liu, L. Huang, B. Wang, and P. Pan, "GeoAug: Data augmentation for few-shot nerf with geometry constraints," in *Comput. Vis.–ECCV 2022: 17th Eur. Conf.*, Berlin, Heidelberg, 2022, pp. 322–337.
[13] C. Li, B. Y. Feng, Z. Fan, P. Pan, and Z. Wang, "StegaNeRF: Embedding invisible information within neural radiance fields," in *2023 IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Paris, France, 2023, pp. 441–453.
[14] A. Chen *et al.*, "MVSNeRF: Fast generalizable radiance field reconstruction from multi-view stereo," in *2021 IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Montreal, QC, Canada, 2021, pp. 14104–14113.
[15] A. Yu, V. Ye, M. Tancik, and A. Kanazawa, "pixelNeRF: Neural radiance fields from one or few images," in *2021 IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Nashville, TN, USA, 2021, pp. 4576–4585.
[16] J. Zhang *et al.*, "Ray priors through reprojection: improving neural radiance fields for novel view extrapolation," in *2022 IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, New Orleans, LA, USA, 2022, pp. 18355–18365.
[17] M. Niemeyer, J. T. Barron, B. Mildenhall, M. S. M. Sajjadi, A. Geiger and N. Radwan, "RegNeRF: Regularizing neural radiance fields for view synthesis from sparse inputs," in *2022 IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, New Orleans, LA, USA, 2022, pp. 5470–5480.
[18] C. Qin and X. Zhang, "Effective reversible data hiding in encrypted image with privacy protection for image content," *J. Vis. Commun. Image Represent.*, vol. 31, no. C, pp. 154–164, 2015. doi: 10.1016/j.jvcir.2015.06.009.
[19] X. Liao and C. Shu, "Reversible data hiding in encrypted images based on absolute mean difference of multiple neighboring pixels," *IEEE Trans. Dependable Secur. Comput.*, vol. 19, no. 2, pp. 992–1002, 2022.
[20] F. Uccheddu, M. Corsini, and M. Barni, "Wavelet-based blind watermarking of 3D models," in *Association for Computing Machinery*. New York, NY, USA, pp. 143–154, 2004.
[21] E. Praun, H. Hoppe, and A. Finkelstein, "Robust mesh watermarking," in *Proc. 26th Annu. Conf. Comput. Graph. Interactive Tech.*, 1999, pp. 49–56.
[22] R. Ohbuchi, A. Mukaiyama, and S. Takahashi, "A frequency-domain approach to watermarking 3D shapes," *Comput. Graph. Forum*, vol. 21, no. 3, pp. 373–382, 2002.
[23] J. U. Hou, D. G. Kim, and H. K. Lee, "Blind 3D mesh watermarking for 3D printed model by analyzing layering artifact," *IEEE Trans. Inf. Forensics Secur.*, vol. 12, no. 11, pp. 2712–2725, 2017. doi: 10.1109/TIFS.2017.2718482.
[24] S. P. Lu, R. Wang, T. Zhong, and P. L. Rosin, "Large-capacity image steganography based on invertible neural networks," in *2021 IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Nashville, TN, USA, 2021, pp. 10811–10820.
[25] J. Jing, X. Deng, M. Xu, J. Wang, and Z. Guan, "HiNet: Deep image hiding by invertible network," in *2021 IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Montreal, QC, Canada, 2021, pp. 4713–4722.
[26] Y. Luo, T. Zhou, F. Liu, and Z. Cai, "IRWArt: Levering watermarking performance for protecting high-quality artwork images," in *Proc. ACM Web Conf. 2023*, Austin, TX, USA, ACM, 2023, pp. 2340–2348.

[27] R. Ma *et al.*, "Towards blind watermarking: Combining invertible and non-invertible mechanisms," in *Proc. 30th ACM Int. Conf. Multimed.*, New York, NY, USA, 2023, pp. 1532–1542.
[28] J. Fridrich, M. Goljan, and R. Du, "Detecting LSB steganography in color, and gray-scale images," *IEEE Multimed.*, vol. 8, no. 4, pp. 22–28, 2001. doi: 10.1109/93.959097.
[29] X. Weng, Y. Li, L. Chi, and Y. Mu, "High-capacity convolutional video steganography with temporal residual modeling," in *Proc. 2019 on Int. Conf. Multimed. Retr.*, New York, NY, USA, 2019, pp. 87–95.
[30] S. G. Mallat, "A theory for multiresolution signal decomposition: The wavelet representation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 11, no. 7, pp. 674–693, 1989. doi: 10.1109/34.192463.
[31] S. Baluja, "Hiding images in plain sight: Deep steganography," in *Proc. 31st Int. Conf. Neural Inf. Process. Syst.*, Red Hook, NY, USA, 2017, pp. 066–2076.
[32] E. Agustsson, "Challenge on single image super-resolution: Dataset and study (supplementary material)," in *2017 IEEE Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Honolulu, HI, USA, 2017, pp. 1122–1131.
[33] V. K. Pallaw, K. U. Singh, A. Kumar, T. Singh, C. Swarup and A. Goswami, "A robust medical image watermarking scheme based on nature-inspired optimization for telemedicine applications," *Electronics*, vol. 12, no. 2, 2023, Art. no. 334. doi: 10.3390/electronics12020334.
[34] A. O. Mohammed, H. I. Hussein, R. J. Mstafa, and A. M. Abdulazeez, "A blind and robust color image watermarking scheme based on DCT and DWT domains," *Multimed. Tools Appl.*, vol. 82, no. 21, pp. 32855–32881, 2023. doi: 10.1007/s11042-023-14797-0.
[35] F. Hu, H. Cao, S. Chen, Y. Sun, and Q. Su, "A robust and secure blind color image watermarking scheme based on contourlet transform and schur decomposition," *Vis. Comput.*, vol. 39, no. 10, pp. 4573–4592, 2023. doi: 10.1007/s00371-022-02610-2.