World Electric Vehicle Journal

MDPI

Article

SRNeRF: Super-Resolution Neural Radiance Fields for Autonomous Driving Scenario Reconstruction from Sparse Views

Jun Wang [1], Xiaojun Zhu [2], Ziyu Chen [1], Peng Li [3], Chunmao Jiang [3], Hui Zhang [2] and Chennian Yu [4] and Biao Yu [3,*]

[1] Science Island Branch, Graduate School of USTC, Hefei 230026, China; wangjun1835568@mail.ustc.edu.cn (J.W.); chenzy1831@mail.ustc.edu.cn (Z.C.)
[2] Jianghuai Advanced Technology Center, Hefei 230088, China; zhu.xj@foxmail.com (X.Z.); gloryzh@mail.ustc.edu.cn (H.Z.)
[3] Hefei Institutes of Physical Science, Chinese Academy of Sciences, Hefei 230031, China; q22301354@stu.ahu (P.L.); sa252@mail.ustc.edu.cn (C.J.)
[4] School of Computer Science and Artificial Intelligence, Chaohu University, Hefei 238024, China; 054071@chu.edu.cn
* Correspondence: byu@hfcas.ac.cn

Abstract: High-fidelity driving scenario reconstruction can generate a lot of realistic virtual simulation environment samples, which can support effective training and testing for autonomous vehicles. Neural radiance fields (NeRFs) have demonstrated their excellence in high-fidelity scenario reconstruction; however, they still rely on dense-view data and precise camera poses, which are difficult to obtain in autonomous vehicles. To address the above issues, we propose a novel approach called SRNeRF, which can eliminate pose-based operations and perform scenario reconstruction from sparse views. To extract more scene knowledge from limited views, we incorporate an image super-resolution module based on a fully convolutional neural network and introduce a new texture loss to capture scene details for higher-quality scene reconstruction. On both object-centric and scene-level datasets, SRNeRF performs comparably to previous methods with ground truth poses and significantly outperforms methods with predicted poses, with a PSNR improvement of about 30%. Finally, we evaluate SRNeRF on our custom autonomous driving dataset, and the results show that SRNeRF can still generate stable images and novel views in the face of sparse views, demonstrating its scalability in autonomous driving scenario synthesis.

Keywords: neural radiance field; autonomous driving scenario reconstruction; sparse views; super-resolution

check for updates

Academic Editor: Michael Fowler

Received: 16 December 2024
Revised: 18 January 2025
Accepted: 20 January 2025
Published: 23 January 2025

Citation: Wang, J.; Zhu, X.; Chen, Z.; Li, P.; Jiang, C.; Zhang, H.; Yu, C.; Yu, B. SRNeRF: Super-Resolution Neural Radiance Fields for Autonomous Driving Scenario Reconstruction from Sparse Views. *World Electr. Veh. J.* **2025**, *16*, 66. https://doi.org/10.3390/wevj16020066

Copyright: © 2025 by the authors. Published by MDPI on behalf of the World Electric Vehicle Association. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

High-fidelity scenario reconstruction can provide extensive realistic environment data for training and testing autonomous vehicles, thereby reducing the requirements of expensive and risky real-world environment samples [1]. Traditional 3D reconstruction methods, such as point-cloud reconstruction and meshes, can only offer rough scene reconstructions and are unable to synthesize new viewpoints or generate high-quality images [2]. With the continuous advancement in deep learning technologies, the advent of neural radiance fields (NeRFs) [3] has addressed this issue. However, they typically require dozens to hundreds of high-quality images with comprehensive coverage from various viewpoints. This reliance on dense views and precise camera poses limits their application in autonomous driving scenarios.

In autonomous driving scenarios, input images are often sparse due to several factors, including limited camera field of view, obstacles in dynamic environments, constraints on sensor configurations, and limitations in data transmission and processing. As a result, only a few images are typically available for each scene. This sparsity leads to a lack of sufficient data and details, leading to inaccuracies [4], overfitting, detail loss [5], consistency issues, and real-time performance problems [6] in scenario reconstruction. These issues impact scenario reconstruction performance for autonomous driving systems. Additionally, autonomous vehicles operate in dynamic and constantly changing environments with complex motion patterns, making it even more difficult to obtain accurate camera poses [7]. The estimation of camera poses relies on various sensors, which are prone to measurement errors, drift, and latency, leading to inaccurate pose information and further impacting the quality of scenario reconstruction.

To address these issues, various methods have emerged in recent years to improve NeRFs. For example, SPARF [1] performs joint pose–NeRF training with a small number of input images and noisy camera pose estimates but relies on dense correspondence networks [8]. In contrast, LEAP [9], as a pose-free method, learns pose-related geometric knowledge and maps from input images to neural volumes from the data themselves. However, the images it generates on small batches or complex datasets exhibit some blurriness. Our method builds on LEAP, improving it by reducing the dependency of neural radiance fields on precise camera poses in sparse views while still achieving satisfactory view synthesis quality. Although our method improves the quality of the synthesized images to some extent, challenges remain when dealing with extremely sparse views and highly dynamic scenes.

Specifically, our main contributions are as follows:

- A super-resolution-based neural radiance field architecture: This architecture enables better reconstruction of autonomous driving scenarios under sparse views. We evaluated the model on progressively complex datasets.
- Detail enhancement module: we incorporated an image super-resolution module, using a fully convolutional neural network, to further improve image clarity and detail.
- Multi-loss optimization: Building on the original L2 loss [10] and perceptual loss [11], we introduced a new texture loss as an additional optimization objective. Texture loss helps the model retain and reconstruct texture details from real images, improving the realism and quality of the synthesized images.

The rest of this paper is organized as follows. In Section 2, we review related work, Section 3 describes our approach, Section 4 presents the experimental results, and Section 5 provides the conclusions.

2. Related Work

2.1. Scenario Reconstruction from Sparse Inputs

Data collected from autonomous vehicle sensors often suffer from single-camera trajectories and limited viewpoints, which constrain the 3D scenario reconstruction capabilities of related NeRF models. This limitation arises because NeRFs' MLP architecture represents each pixel of the input image with separate neurons and requires independent optimization for each scenario, meaning that knowledge cannot be shared between scenarios and any prior knowledge about the scene cannot be utilized for reconstruction [3]. When the number of captured viewpoints is insufficient, the reconstruction quality rapidly declines or even fails. In NeRF methods, hundreds or even thousands of multi-view images are used, and this is highly resource-intensive.

Considerable work has been proposed to solve these problems. Yu et al. [12] proposed PixelNeRF, which simulates continuous neural scenes using only a few input images. It builds on the NeRF by incorporating convolutional neural networks and bilinear interpola-

tion pretraining layers to extract image features. For each sampled point, corresponding image features are extracted, making full use of the input image features. These features, along with the feature points, spatial locations, and viewpoint directions, are fed into the NeRF network, allowing for the construction of continuous static scenes from a sparse set of images. Similarly, Trevithick et al. [13] introduced the Genera Radiance Field (GRF), which uses a similar approach to PixelNeRF, but operates in a standard canonical space rather than the camera coordinate system, making it more general and versatile. Additionally, Chen et al. [14] proposed MVSNeRF, which leverages the strengths of deep learning-based multi-view stereo and neural rendering to synthesize novel viewpoints from three input viewpoints. Xu et al. [15] introduced Point-NeRF, which combines point-cloud methods in explicit representation with neural radiance field techniques in implicit representation. This approach encodes the geometric shapes and appearance of the scene using point clouds and incorporates point-cloud clipping algorithms to avoid sampling invalid areas, effectively reducing the number of sampled points.

In recent advancements, Chen et al. [16] proposed 3DTopia-XL as a scalable solution for generating high-quality 3D assets. 3DTopia-XL utilizes a diffusion transformer (DiT)-based generative framework to produce high-quality 3D assets from minimal inputs. However, this framework has not been applied to sparse input-based autonomous driving scenario reconstruction.

2.2. NeRF Under Noisy Camera Poses

Autonomous vehicles operate in dynamic environments where scenarios and objects are constantly changing, and their motion patterns are complex. These complex motion patterns increase the difficulty of camera pose estimation. Building NeRFs from images without accurate camera poses is challenging. To address this issue, NeRS [17] uses surface-based 3D representations as optimization regularizers, but this approach is primarily suited to object-centric data. SPARF [1] uses pretrained dense correspondence networks to extract matches between training views and applies multi-view correspondence loss to minimize reprojection errors between matches for scene and pose estimation. However, its effectiveness largely depends on the accuracy of dense correspondences and the initial poses. Regarding generalizable NeRF variants, SRT [2] proposes a pose-free paradigm to build 2D latent scene representations for driving scenes, but SRT lacks 3D perception capabilities, resulting in limited-quality novel view synthesis. FORGE [18] jointly estimates camera poses and predicts radiance fields; however, its performance is highly sensitive to pose estimation accuracy, and it faces challenges in multi-stage training. In contrast, LEAP [9] employs a feature similarity-driven 2D–3D information mapping approach, which eliminates the dependence on camera poses and produces results closer to those obtained with ground truth poses.

2.3. Image Super-Resolution

Image super-resolution is an important image processing technique in computer vision, which involves recovering high-resolution images from low-resolution ones. When NeRF-generated images lack detail due to sparse inputs, SR models can expand the limited low-resolution information into high-resolution details. This process allows the model to learn more scenario knowledge from limited views, compensating for the effects of insufficient viewpoints. By combining NeRFs with SR models, the strengths of both can be fully utilized, enabling the creation of high-quality images even with sparse inputs.

With the advancement of deep learning technologies, particularly convolutional neural networks (CNNs), significant progress has been made in image super-resolution. Dong et al. [19] applied shallow networks to the task of super-resolution image reconstruction, training CNNs via backpropagation to learn to map from a low-resolution input with

bicubic interpolation to high-resolution images. In addition to training by minimizing pixel-wise loss functions (such as the mean-squared error), some methods have introduced generative adversarial networks (GANs) [20], further enhancing the quality of the images generated. Meanwhile, the IPG proposed by Tian et al. [21] is an image super-resolution method based on graph neural networks (GNNs), which breaks the rigidity of traditional methods and enhances super-resolution performance by leveraging the flexibility of graphs.

In scene reconstruction tasks, image super-resolution technology plays a critical and indispensable role. Scene reconstruction often requires generating complete 3D scenes or high-resolution images from sparse viewpoints, and super-resolution can effectively mitigate the detail loss caused by low-resolution images. In scene reconstruction based on multi-view geometry, employing super-resolution techniques can significantly enhance the clarity of texture details, thereby improving the visual fidelity of the reconstruction model.

3. Materials and Methods

In this section, we introduce our method for synthesizing novel views from pose-free sparse view inputs. The network architecture, illustrated in Figure 1, consists of three components, the image rendering module, the upsampling module, and the loss function.

During the training phase, input images were first processed by the ViT [22] network to extract image features. We selected the first image from the image set as the canonical view. The image features were then passed through a multi-view encoder to enhance the consistency of information representation between the canonical view and non-canonical views. The updated feature 2D representation was then remapped to the 3D domain of neural volume representation and finally rendered into an RGB image through neural radiance field rendering. This image was fed into an upsampling module consisting of a series of convolutional layers for refinement, and the final output was compared with the input image.

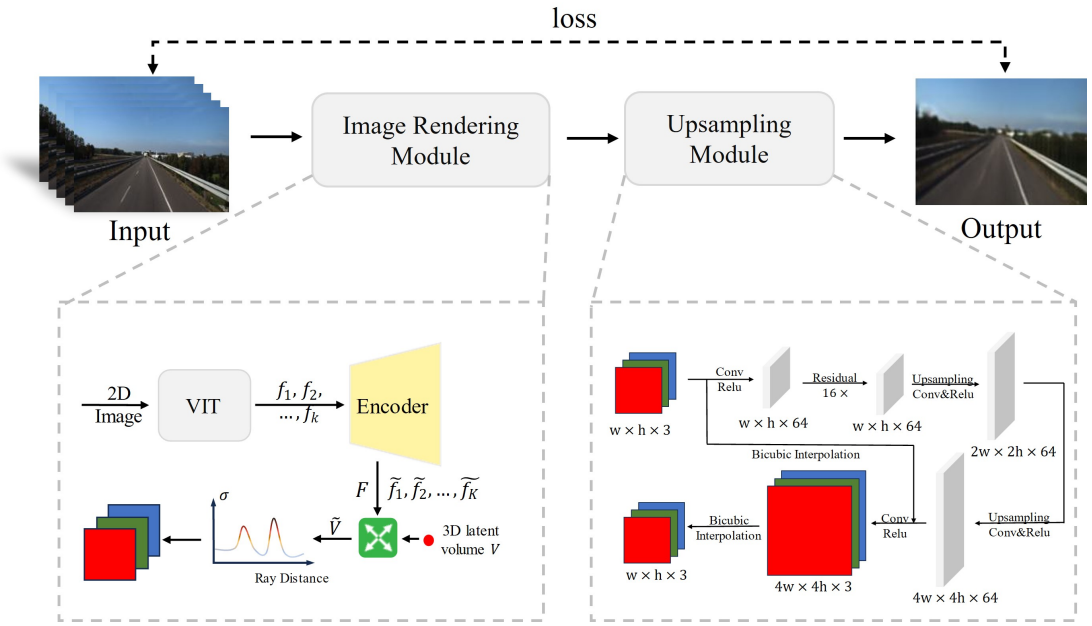


Figure 1. The network architecture. It consists of three components: the image rendering module, the upsampling module, and the loss function.

3.1. Image Rendering Module

As illustrated in Figure 2, the architecture of the image rendering module consists of four parts.

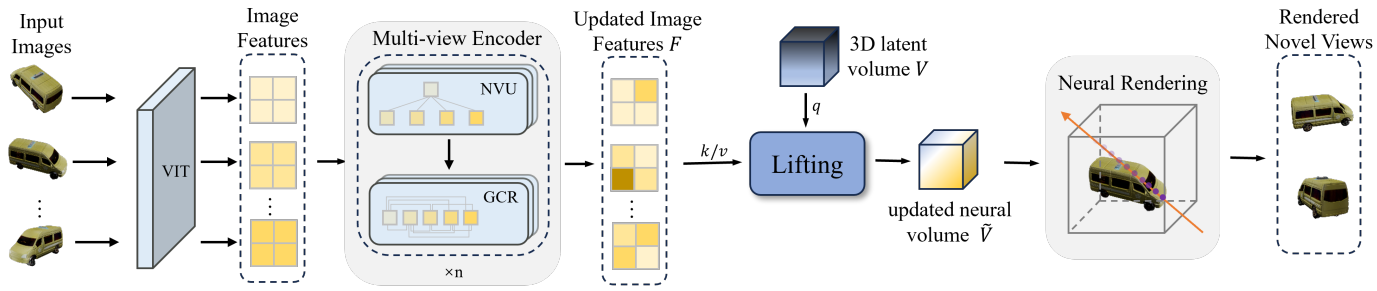


Figure 2. Architecture of image rendering module. Starting from the input image features, feature extraction and processing are performed through the multi-view encoder. The neural volume data are then updated, and finally, the neural rendering module generates an image of the new view.

Feature extraction: This network extracts image features:

$$f_i \in \mathbb{R}^{h \times w \times c},\ i = 1, \ldots, k \tag{1}$$

where h, w, c, and k represent the image height, width, and number of channels and the number of input images, respectively. This was carried out to capture the diversity and richness of the scene. We chose to use ViT initialized with DINOv2 [23] as the feature extractor primarily due to its impressive performance in handling image data.

DINOv2 is a self-supervised learning model, and its feature extraction process is based on self-attention mechanisms, allowing the model to establish effective correlations between different regions and capture global context information. Through training on a large-scale unlabeled image dataset, ViT can effectively learn rich feature representations of images. In the image rendering module, we aimed to leverage the cross-view correlation modeling ability of the ViT model to better understand scene information from different viewpoints [24].

The multi-view encoder: This consists of n blocks, each containing two layers, which are the non-canonical view update (NVU) layer and the global consistency reasoning (GCR) layer [9].

The NVU layer is a Transformer layer with cross-attention, where non-canonical view features serve as queries and canonical view features serve as keys/values. The NVU layer updates the features of each non-canonical view by aggregating canonical view features. This is represented by the following equation:

$$\bar{f}_j = NVU(f_j, f_1), \quad j \neq 1 \tag{2}$$

$$[\bar{f}_2, \ldots, \bar{f}_k] = FFN(\text{CrossAttention}([f_2, \ldots, f_k], f_1)) \tag{3}$$

where $\bar{f}_j$ represents the features after the NVU layer. $[\bar{f}_2, \ldots, \bar{f}_k] \in \mathbb{R}^{(k-1)hw \times c}$, and $f_1 \in \mathbb{R}^{hw \times c}$.

The GCR layer is a Transformer layer with self-attention, where the features outputted by the NVU layer serve as queries, keys, and values. The GCR layer leverages the correlations between all views to perform global consistency reasoning across all views. This is represented by the following equation:

$$\tilde{f}_k = GCR(f_1, \bar{f}_2, \ldots, \bar{f}_k) \tag{4}$$

$$[\tilde{f}_1, \ldots, \tilde{f}_k] = FFN(\text{SelfAttention}([f_1, \bar{f}_2, \ldots, \bar{f}_k])) \tag{5}$$

where $\tilde{f}_k$ represents the features after the GCR layer. For simplicity, we denote the output of the multi-view encoder $[\tilde{f}_1, \ldots, \tilde{f}_k]$ as F.

Two-dimensional–three-dimensional lifting: Firstly, we introduce a 3D latent neural volume $V \in \mathbb{R}^{H \times W \times D \times c}$, where V is defined in the local camera coordinate system of the

canonical views. This part utilizes n_m Transformer decoding blocks, where each decoding block consists of a cross-attention layer and a self-attention layer.

In the cross-attention layer, V serves as the queries, while the output of the multi-view encoder F serves as the keys/values. For each 3D point in the neural volume V, we compute its feature similarity with all 2D image features in F and use this similarity to obtain a weighted average of the 2D image features. This allows the features of each 3D point to integrate information from multiple viewpoints, thereby better reflecting the characteristics of the entire scene. This is represented by the following equation:

$$\bar{V} = \text{CrossAttention}(V, F) \tag{6}$$

The self-attention layer refines the updated 3D volume features $\bar{V}$. The self-attention mechanism allows each voxel to consider the relationships with other voxels, thus capturing long-range geometric correlations more effectively. This refinement process helps to further enhance the representation capability of the neural volume, enabling it to more accurately encode the geometric structure and texture features of the scene. The updated neural volume representation is given by $\tilde{V} \in \mathbb{R}^{H \times W \times D \times c}$.

Neural rendering: Firstly, the updated neural volume $\tilde{V}$ [25] is predicted with $R = (R_\sigma, R_f)$ through 3D convolutional layers, where R_σ and R_f are the density and feature of the radiance field, respectively. This process is represented as:

$$R_\sigma, R_f = \text{Conv3D}(\tilde{V}) \tag{7}$$

where $R_\sigma \in \mathbb{R}^{H' \times W' \times D'}$ and $R_f \in \mathbb{R}^{H' \times W' \times D' \times C}$. H', W', and D' are the spatial resolutions. Then, employing volume rendering techniques[3], the process is represented as follows:

$$(\hat{I}, \hat{\sigma}) = \Pi(R, \Phi) \tag{8}$$

where Π represents the volume rendering process, Φ is the target camera pose, and the output includes the rendered image $\hat{I}$ and object mask $\hat{\sigma}$.

3.2. Upsampling Module

The output of the image rendering module, the initially rendered RGB image, serves as the input to the upsampling module, which refines the rendered image.

The module we added, inspired by Lee et al.'s work [26], employs a fully convolutional neural network architecture. As shown in Figure 3, during the forward pass, the input image first passes through a convolutional layer, followed by a ReLU activation function. It then enters the Generator network, which consists of convolutional layers, ReLU activations, and multiple residual blocks. Each residual block contains two convolutional layers that learn the residuals of the input feature maps to better preserve image details. Finally, an upsampling layer is used to enlarge the feature maps.

To mitigate checkerboard artifacts [27] caused by transposed convolution layers, a convolutional layer was added after the upsampling blocks to avoid regular patterns in the output. Image enhancement was then performed, with Kim et al.'s [28] approach of adding bicubic interpolation to the model's output, which helps stabilize training and reduce color shifts in the output during training.

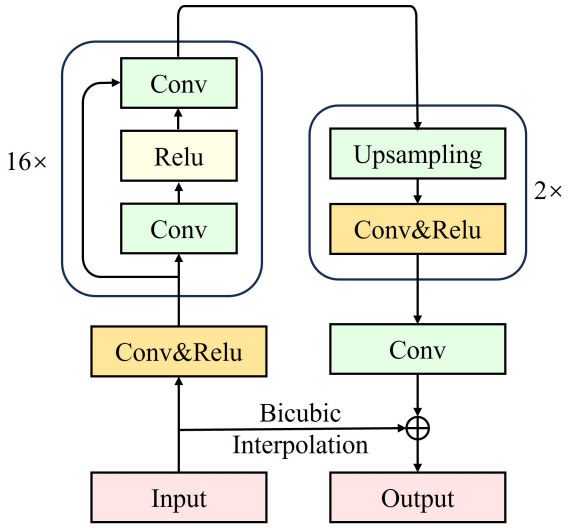


Figure 3. Network architecture of the upsampling module. It consists of convolutional layers, several Generator networks, and two upsampling layers. The image goes through a series of convolutions (with 3 × 3 kernels) and ReLU operations, followed by an upsampling step, and finally, bicubic interpolation is applied to increase the resolution.

3.3. Combined Loss Function

We define the joint loss L applied to the network as follows:

$$L = L_I + \lambda_m L_M \tag{9}$$

$$L_I = \sum_i L_{\text{mse}}(\hat{I}_i, I_i) + \sum_i \lambda_p L_p(\hat{I}_i, I_i) + \sum_i \lambda_t L_t(\hat{I}_i, I_i) \tag{10}$$

$$L_M = \sum_i L_{\text{mse}}(\hat{\sigma}_i, \sigma_i) \tag{11}$$

where L_I is the loss applied to the image, L_M is the loss applied to the density mask, $\hat{I}_i$ and I_i are the predicted and ground truth images, respectively, $\hat{\sigma}_i$ and σ_i are the original and rendered density masks, respectively, and λ_p and λ_t are hyperparameters used to balance the losses.

The L2 loss [10] measures the difference by computing the squared difference between the predicted values and the ground truth values. It performs well in many cases but also has some limitations. For example, it is not sensitive to high-frequency details, it is unable to effectively capture subtle differences, and it tends to produce smooth outputs, leading to blurry results. This perceptual loss helps improve structural similarity and perceptual quality by focusing on higher-level features, but it still struggles to preserve local texture details. In contrast, texture loss can capture local details and texture information in the image, thereby generating more realistic and detailed images. Therefore, we introduced texture loss, L_t, as an additional optimization objective, in addition to the existing L2 loss L_{mse} and perceptual loss L_p [11].

We implemented a PyTorch module named VGGTextureLoss, which was used to compute the texture loss of an image. This module is based on the feature extractor of the VGG19 model. VGG19 [29] is a deep convolutional neural network that has been extensively trained and has learnt feature representations with good generalization and abstraction properties. Its structure is relatively simple and computationally lightweight.

We input both the predicted image, $\hat{I}_i$, and the original image, I_i, into the VGG19 network to extract features at a specified layer (default to conv4_2). Let $\phi(I) \in \mathbb{R}^{n \times m}$ denote the extracted features, and we calculated the mean-squared error of the Gram

matrix $G(\phi(I))$ between these features as the texture loss, as shown in the formula, where $G(F) = FF^T \in \mathbb{R}^{n \times n}$.

$$L_t = \|G(\phi(I_i)) - G(\phi(\hat{I}_i))\|_2^2 \quad (12)$$

4. Results

4.1. Datasets

We evaluated the performance of our proposed model on the following datasets:

- OmniObject3D [30]: This is a large-scale real-object dataset with rich textures, though the geometric shapes are relatively simple compared to other datasets. The objects in this dataset cover a variety of categories, including different shapes and materials.
- Kubric-ShapeNet [18]: Created by Jiang et al. in 2022, this dataset's test set consists of two parts: (i) 1300 object instances from training categories and (ii) 1000 object instances from 10 novel object categories. This allows researchers to assess a model's performance across different and unseen categories, providing a comprehensive evaluation of reconstruction quality and generalization capability.
- DTU: this is a small-scale dataset that contains image sequences of various scenes, each captured from multiple viewpoints.
- KITTI: This is a widely used large-scale dataset for autonomous driving and computer vision research. We selected 815 typical scenes from it, covering a range of road conditions, including urban, rural, and highway environments, to validate the algorithm's robustness in diverse autonomous driving scenarios.
- Waymo [31]: this is one of the largest and most advanced autonomous driving datasets available. We selected 825 scenes from it, with data collected from various geographical locations, offering high diversity and representativeness.
- Custom Dataset Science Island: we created our own dataset by capturing scenes on Science Island, including 125 scenes covering various weather and lighting conditions.

4.2. Evaluation Metrics

- The PSNR (peak signal-to-noise ratio) is a commonly used metric for measuring image quality. It quantifies the distortion of an image by comparing the mean-squared error between the original and reconstructed images. A higher PSNR value indicates lower distortion and better image quality. However, the PSNR does not take into account human perceptual characteristics and is less sensitive to image details and structural information. Therefore, in certain cases, particularly with structural changes in complex images, the PSNR may not fully reflect changes in image quality.
- The SSIM (structural similarity index) is a structured image quality assessment metric. Unlike the PSNR, the SSIM considers not only brightness information but also structural information about the image. Therefore, the SSIM can better reflect humans' perception of the image. An SSIM value closer to 1 indicates higher similarity between the reconstructed and original images. We chose the SSIM as an auxiliary evaluation metric primarily because it can more accurately measure structural similarity, especially in images with rich details. However, the SSIM also has limitations, as its performance may not meet expectations for low-contrast images or localized regions.
- The LPIPS (perceptual L2 distance) [32] is a perceptual image similarity metric. Unlike the traditional L2 distance, the LPIPS captures perceptual differences between images more effectively. A lower LPIPS value indicates a smaller perceptual distance between the reconstructed and original images, indicating higher similarity between the images. The LPIPS was selected as the third evaluation metric because it directly simulates

human perception through deep network models, providing a better reflection of the perceptual quality of images.

4.3. Implementation Details

Our experiments were conducted on an NVIDIA A800 GPU (AMAX China, Suzhou, China), and the operating system used was Ubuntu 20.04. The system was powered by an AMD EPYC 7402 24-core processor (AMAX China, Suzhou, China), featuring an x86_64 architecture. Our experimental setup is summarized in Table 1:

Table 1. Experimental settings used in our study.

Parameter	Setting
Number of views	5
Image resolution	224×224
Weights	$\lambda_p = \lambda_t = 0.1, \quad \lambda_m = 5.0$
Initial LR (backbone)	2×10^{-5}
Initial LR (others)	2×10^{-4}
Scheduler warmup iter	500
Optimizer	AdamW
Encoder blocks	2
Lifting blocks	4
3D volume resolution	$163 \times 163 \times 163$
Radiance field resolution	$64 \times 64 \times 64$
Points per ray	64

4.4. Object-Centric Results

On object-centric datasets, SRNeRF demonstrates strong modeling capabilities, as shown in Table 2. The optimal results have been highlighted in bold. SRNeRF achieves outstanding results across all three datasets, particularly excelling in the PSNR metric. Compared to methods without pose information, SRNeRF not only surpasses the current best baselines but also shows significant improvement over SRT.

Table 2. Evaluation on three object-centric test sets.

Model	Pose	Inf. Time	Omniobject3D			Kubric-ShapeNet-Seen			Kubric-ShapeNet-Novel		
			PSNR ↑	SSIM ↑	LPIPS ↓	PSNR ↑	SSIM ↑	LPIPS ↓	PSNR ↑	SSIM ↑	LPIPS ↓
PixelNeRF	GT	2 min	26.97	0.888	0.123	29.25	0.893	0.127	29.37	0.906	0.122
	Pred		18.87	0.810	0.199	21.36	0.836	0.188	21.22	0.851	0.174
SRT	×	0.4 s	20.22	0.786	0.303	22.62	0.802	0.267	22.46	0.793	0.284
LEAP	×	0.5 s	28.10	0.897	0.101	29.86	0.926	0.067	28.22	0.924	0.070
SRNeRF	×	0.5 s	**28.58**	**0.908**	**0.082**	**29.89**	**0.930**	**0.065**	**29.59**	**0.926**	**0.068**

We further tested the model's generalization capability using the Kubric-Shapenet dataset. The results show that SRNeRF not only performs exceptionally well on the training data but also demonstrates strong generalization ability when handling novel data. Additionally, in terms of runtime, SRNeRF performs comparably to LEAP and outperforms other comparable methods. Figure 4 shows the comparison results between our method and other approaches.

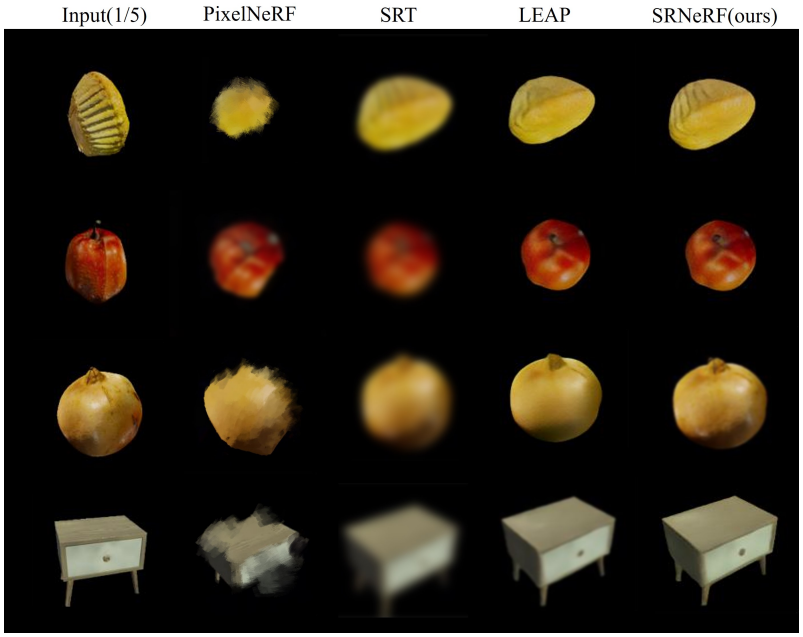


Figure 4. Comparison with prior methods. PixelNeRF quickly deteriorates under predicted poses, and SRT can only produce blurry images. LEAP can synthesize high-quality images, but its handling of lighting and shadows lacks realism. In contrast, SRNeRF reliably recovers details, with the novel views closely matching the ground truth target views.

4.5. Scenario Reconstruction Results

In our research, we applied the model to autonomous driving scenarios and conducted experiments on the KITTI dataset, the Waymo dataset, and our self-created dataset, Science Island. In autonomous driving applications, due to complex and ever-changing road environments, the input images are typically very sparse, with only a few images available for each scene. This poses a significant challenge to the model's ability to learn the scene. The model needs not only to extract useful information from these limited images but also to synthesize images from different perspectives to form a comprehensive understanding of the scene. To address this challenge, we set up our experiments with only five randomly selected images as the input each time, rigorously testing the model's performance under sparse image conditions.

As shown in Table 3, the optimal results are highlighted in bold. SRNeRF achieved better metric values than LEAP in our experiments, with a more significant improvement on the Science Island dataset, where the PSNR increased by approximately 20%.

Table 3. Evaluation on street views. Comparison of PixelNeRF, LEAP, and SRNeRF across three datasets shows that SRNeRF achieves better metric results.

Model	Pose	Inf. Time	KITTI			Waymo Open Dataset			Science Island		
			PSNR ↑	SSIM ↑	LPIPS ↓	PSNR ↑	SSIM ↑	LPIPS ↓	PSNR ↑	SSIM ↑	LPIPS ↓
PixelNeRF	GT	3 min	10.83	0.233	0.643	14.78	0.317	0.658	15.52	0.511	0.563
	$\sigma = 0.15$		9.01	0.178	0.735	12.88	0.281	0.730	13.27	0.493	0.665
LEAP	×	0.5 s	13.04	0.348	0.559	16.37	0.456	0.564	17.76	0.641	0.412
SRNeRF	×	0.6 s	14.01	0.372	0.552	18.97	0.476	0.551	**21.65**	**0.658**	**0.396**
SRNeRF-pretrained	×	0.4 s	**14.88**	**0.378**	**0.532**	**19.33**	**0.501**	**0.534**		-	

Additionally, we included a comparison with PixelNeRF. When the camera pose information in the street view was contaminated by noise ($\sigma = 0.15$), PixelNeRF failed,

whereas our method, which does not rely on precise camera poses, still achieved stable results. This characteristic endows SRNeRF with higher robustness and adaptability in practical applications.

We compare with LEAP in Figures 5 and 6. When processing street scene inputs, LEAP exhibits significant blurriness and artifacts, resulting in poor reconstruction quality. In contrast, SRNeRF can better restore road conditions, generating clearer images with more complete detail preservation. This indicates that SRNeRF has higher stability and accuracy in handling sparse view reconstruction tasks. Especially in applications like autonomous driving, where high image quality is crucial, SRNeRF can provide more reliable visual information.

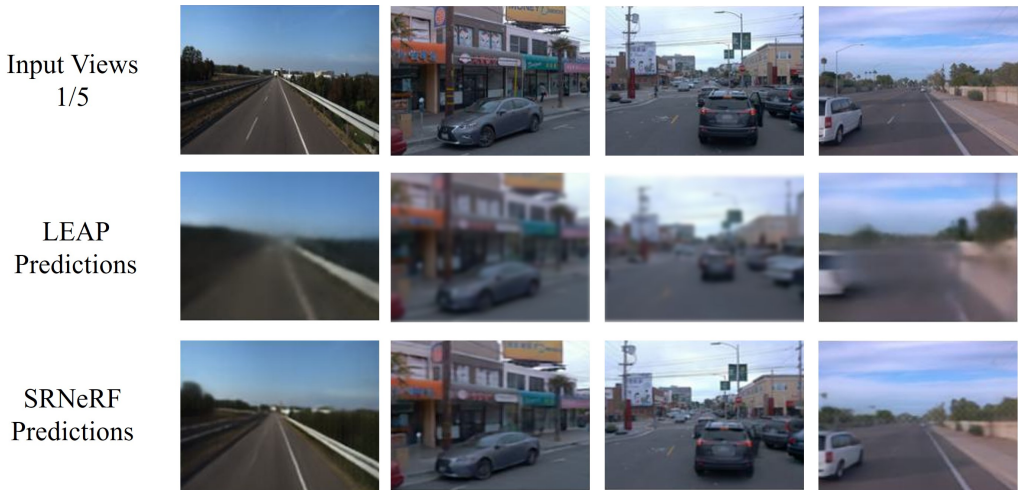


Figure 5. Qualitative results on KITTI (first column) and Waymo datasets (second to fourth columns). PixelNeRF quickly deteriorates under predicted poses, and SRT can only produce blurry images. LEAP can synthesize high-quality images, but its handling of lighting and shadows lacks realism. In contrast, SRNeRF reliably recovers details, with the novel views closely matching the ground truth target views.

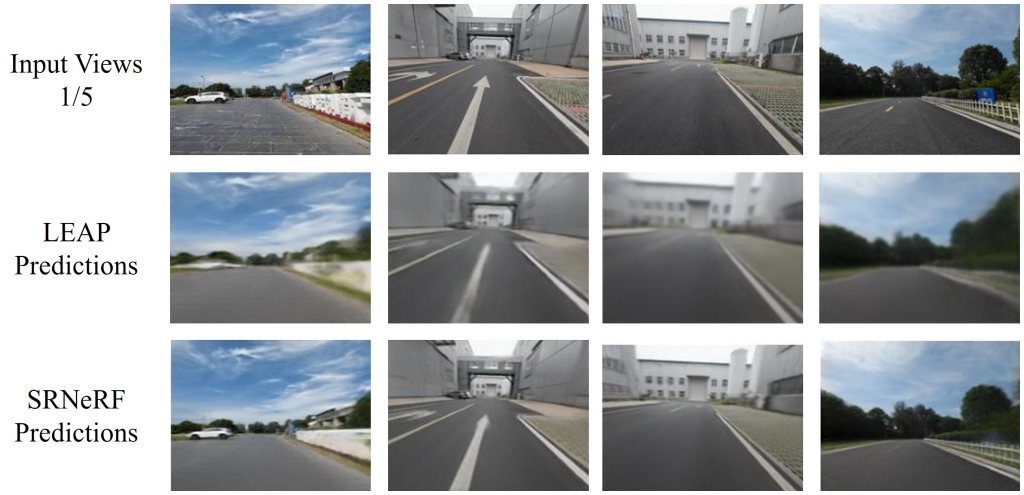


Figure 6. Qualitative results on Science Island dataset. On our own dataset, SRNeRF demonstrates superior metrics and better synthesis results.

4.6. New View Synthesis Results

Figure 7 showcases the new view synthesis results of LEAP and SRNeRF on the KITTI and Waymo datasets. The first row in the figure presents the input ground truth images, the second row displays the images generated by LEAP, and the third row shows the images produced by our SRNeRF model.



Figure 7. Visualization of NVS results. The images showcase the new view synthesis performance of LEAP and SRNeRF on the KITTI and Waymo datasets.

The images generated by LEAP exhibit several shortcomings in new view synthesis. While the contours of vehicles and buildings can be roughly identified, the overall clarity is poor, and details are difficult to recover. The images generally suffer from significant blurring, leading to a loss of detail and texture information. Additionally, the LEAP-generated images exhibit considerable distortion and noise, resulting in a less-than-ideal visual effect that fails to present a realistic scene impression.

In contrast, the SRNeRF model demonstrates superior detail retention and visual clarity in new view synthesis tasks. With the same sparse view inputs, SRNeRF more accurately reconstructs the contours of vehicles and buildings, and the image clarity is significantly improved. Especially in terms of texture and detail, SRNeRF's performance is exceptional, with almost no noticeable distortion or noise. The positions and layouts of objects in the scene are also more accurately reconstructed, resulting in more realistic new view images. These advantages make SRNeRF more feasible and practical for applications in autonomous driving scenarios.

4.7. Ablation Study

We conducted ablation experiments on the texture loss and super-resolution (SR) model, with the results shown in Table 4. The optimal results have been highlighted in bold. It can be seen that the presence or absence of texture loss has a more significant impact on perceptual quality metrics like the SSIM. This is because texture loss captures more details, improving the quality and realism of the novel views.

At the same time, the experiments also show that removing the SR model results in a more noticeable decline in metrics like the PSNR, highlighting the indispensable role of the SR model in SRNeRF. The SR model performs super-resolution processing on the input images, learning more scene knowledge and thus playing a crucial role in the sparse view reconstruction task. Overall, the combined effect of texture loss and the SR model enables SRNeRF to excel in generating high-quality novel views, significantly outperforming baseline models that do not include these modules. This further validates the effectiveness and rationale of our method's design.

Additionally, as an essential component of the image rendering module, we explored the impact of using the multi-view encoder. Without this module, the model's performance significantly deteriorated due to failure in terms of view updating and global consistency reasoning during the image rendering process, leading to a substantial decline in image reconstruction quality. This result indicates that the multi-view encoder is crucial for view consistency and feature integration.

Table 4. Ablation study results. The results show that removing the texture loss or the SR module leads to worse performance on these metrics, indicating the crucial role of our improved modules.

Model	DTU		
	PSNR ↑	SSIM ↑	LPIPS ↓
SRNeRF (full)	**16.35**	**0.493**	**0.431**
SRNeRF (without texture loss)	15.83	0.454	0.458
SRNeRF (without SR module)	15.12	0.428	0.462
SRNeRF (without multi-view enc.)	12.80	0.327	0.498

5. Discussion

This study focuses on autonomous driving scenario reconstruction from sparse views and proposes a scene synthesis method called SRNeRF. This method combines a neural radiance field-based image rendering module with an upsampling module. Additionally, a joint loss function is introduced to improve the quality of scenario reconstruction. To validate the effectiveness of the proposed method, we conducted experiments on multiple datasets and compared our model with relevant algorithms.

First, we validated the model's modeling capability on the OmniObject3D and Kubric datasets. One of the key strengths of SRNeRF is its ability to maintain rendering quality even when dealing with noisy poses or in the absence of ground truth poses. This advantage is particularly evident in comparison with PixelNeRF. While PixelNeRF performs well with ground truth poses, its performance significantly deteriorates with predicted poses. In contrast, SRNeRF is not affected by these limitations and consistently delivers better results even in challenging conditions. In addition, the runtime indicates that SRNeRF not only has an advantage in terms of accuracy but is also competitive in terms of efficiency.

Next, we applied the model to autonomous driving scenarios, and SRNeRF achieved better metric values compared to the baseline methods. To verify the generalization capability of SRNeRF, we applied SRNeRF, pretrained on the Science Island dataset, to the KITTI and Waymo datasets. The results show that pretraining SRNeRF not only led to improved metrics but also reduced the inference time from 0.6 s to 0.4 s. This is because the pretrained model had already learned general features from the driving scene dataset, making feature extraction more efficient. Compared to models trained from scratch, the pretrained model could capture the key features of the input data more quickly, thereby reducing the inference time. The effectiveness of pretraining demonstrates the model's ability to learn general scene knowledge, which is transferable and applicable to new domains.

We also conducted new view synthesis experiments, where the SRNeRF model demonstrated superior detail retention and visual clarity in these tasks. Finally, we performed ablation experiments to validate the importance of texture loss, the super-resolution module, and the multi-view encoder to the model.

In summary, our method achieved better results than the comparative methods across multiple datasets, but its performance in extremely complex and highly dynamic environments still requires further investigation, which will be a focus of future research.

6. Conclusions

Our research on SRNeRF has successfully improved new view synthesis technology based on the LEAP method, especially in the context of sparse views. By introducing a detail enhancement module and a multi-loss optimization strategy, we have reduced the reliance on precise camera poses, significantly decreased the risk of overfitting, and addressed consistency issues in the synthesized images. As a result, our method enhances the clarity, realism, and detail representation of the synthesized images. Extensive evaluation

on diverse datasets further confirms the superiority and scalability of our approach across various complex scenarios. In autonomous driving simulation, our method can generate higher-quality environmental images, thereby enhancing the perception system's understanding of the surrounding environment and improving the accuracy of decision-making and planning.

Although our method has shown significant improvements, there are still challenges when dealing with extremely sparse views and highly dynamic scenes. Additionally, as a learning-based approach, more training typically leads to better performance. Future work will continue to explore how to better leverage input views, achieve improved results with less training data, and apply the method to complex scenarios such as autonomous driving to meet growing technical demands and challenges.

Author Contributions: Conceptualization, J.W. and B.Y.; methodology, J.W. and B.Y.; software, J.W. and X.Z.; validation, J.W. and Z.C.; formal analysis, J.W. and P.L.; investigation, J.W. and C.J.; data curation, C.Y. and H.Z.; writing—original draft preparation, J.W.; writing—review and editing, B.Y.; visualization, J.W.; supervision, B.Y.; project administration, X.Z. and H.Z.; funding acquisition, B.Y. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by the Youth Innovation Promotion Association of the CAS under Grant 2021115, Dreams Foundation of Jianghuai Advance Technology Center (No. 2023-ZM01G002).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The original contributions presented in this study are included in the article. Further inquiries can be directed to the corresponding author.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Truong, P.; Rakotosaona, M.J.; Manhardt, F.; Tombari, F. Sparf: Neural radiance fields from sparse and noisy poses. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Vancouver, BC, Canada, 17–24 June 2023; pp. 4190–4200.
2. Sajjadi, M.S.; Meyer, H.; Pot, E.; Bergmann, U.; Greff, K.; Radwan, N.; Vora, S.; Lučić, M.; Duckworth, D.; Dosovitskiy, A. Scene representation transformer: Geometry-free novel view synthesis through set-latent scene representations. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 6229–6238.
3. Mildenhall, B.; Srinivasan, P.P.; Tancik, M.; Barron, J.T.; Ramamoorthi, R.; Ng, R. Nerf: Representing scenes as neural radiance fields for view synthesis. *Commun. ACM* **2021**, *65*, 99–106. [CrossRef]
4. Du, X.; Sun, H.; Lu, M.; Zhu, T.; Yu, X. DreamCar: Leveraging Car-specific Prior for in-the-wild 3D Car Reconstruction. *arXiv* **2024**, arXiv:2407.16988. [CrossRef]
5. Tu, J.; Ji, W.; Zhao, H.; Zhang, C.; Zimmermann, R.; Qian, H. DriveDiTFit: Fine-tuning Diffusion Transformers for Autonomous Driving. *arXiv* **2024**, arXiv:2407.15661.
6. Lange, B.; Itkina, M.; Li, J.; Kochenderfer, M.J. Self-supervised Multi-future Occupancy Forecasting for Autonomous Driving. *arXiv* **2024**, arXiv:2407.21126.
7. Lin, C.H.; Ma, W.C.; Torralba, A.; Lucey, S. Barf: Bundle-adjusting neural radiance fields. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, QC, Canada, 10–17 October 2021; pp. 5741–5751.
8. Truong, P.; Danelljan, M.; Van Gool, L.; Timofte, R. Learning accurate dense correspondences and when to trust them. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 5714–5724.
9. Jiang, H.; Jiang, Z.; Zhao, Y.; Huang, Q. Leap: Liberate sparse-view 3d modeling from camera poses. *arXiv* **2023**, arXiv:2310.01410.
10. Wang, Y.; Zhou, C.; Qi, X.; Li, H. UHRNet: a deep learning-based method for accurate 3D reconstruction from a single fringe-pattern. *J. Mod. Opt.* **2023**, *70*, 707–722. [CrossRef]
11. Kashkarov, E.; Chistov, E.; Molodetskikh, I.; Vatolin, D. Can No-Reference Quality-Assessment Methods Serve as Perceptual Losses for Super-Resolution? *arXiv* **2024**, arXiv:2405.20392.

12. Yu, A.; Ye, V.; Tancik, M.; Kanazawa, A. pixelnerf: Neural radiance fields from one or few images. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 4578–4587.
13. Trevithick, A.; Yang B. Grf: Learning a general radiance field for 3d representation and rendering. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, QC, Canada, 10–17 October 2021; pp. 15182–15192.
14. Chen, A.; Xu, Z.; Zhao, F.; Zhang, X.; Xiang, F.; Yu, J.; Su, H. Mvsnerf: Fast generalizable radiance field reconstruction from multi-view stereo. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, QC, Canada, 10–17 October 2021; pp. 14124–14133.
15. Xu, Q.; Xu, Z.; Philip, J.; Bi, S.; Shu, Z.; Sunkavalli, K.; Neumann, U. Point-nerf: Point-based neural radiance fields. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 5438–5448.
16. Chen, Z.; Tang, J.; Dong, Y.; Cao, Z.; Hong, F.; Lan, Y.; Wang, T.; Xie, H.; Wu, T.; Saito, S.; Pan, L. 3dtopia-xl: Scaling high-quality 3d asset generation via primitive diffusion. *arXiv* **2024**, arXiv:2409.12957.
17. Zhang, J.; Yang, G.; Tulsiani, S.; Ramanan, D. Ners: Neural reflectance surfaces for sparse-view 3d reconstruction in the wild. *Adv. Neural Inf. Process. Syst.* **2021**, *34*, 29835–29847.
18. Jiang, H.; Jiang, Z.; Grauman, K.; Zhu, Y. Few-view object reconstruction with unknown categories and camera poses. In Proceedings of the 2024 International Conference on 3D Vision (3DV), Davos, Switzerland, 18–21 March 2024; IEEE: Piscataway, NJ, USA, 2024; pp. 31–41.
19. Zheng, H.; Hakim, L.; Kurita, T.; Miyao, J. Single-Image Super-Resolution Reconstruction Based on the Differences of Neighboring Pixels. In *Proceedings of the Neural Information Processing: 28th International Conference, ICONIP 2021, Sanur, Bali, Indonesia, 8–12 December 2021* Proceedings, Part V 28; Springer International Publishing: Berlin/Heidelberg, Germany, 2021; pp. 522–529.
20. Aghelan, A.; Amiryan, A.; Zarghani, A.; Hatami, B.; Rouhani, M. IG-CFAT: An Improved GAN-Based Framework for Effectively Exploiting Transformers in Real-World Image Super-Resolution. *arXiv* **2024**, arXiv:2406.13815.
21. Tian, Y.; Chen, H.; Xu, C.; Wang, Y. Image Processing GNN: Breaking Rigidity in Super-Resolution. In Proceedings of the 2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 16–22 June 2024; pp. 24108–24117.
22. Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv* **2020**, arXiv:2010.11929.
23. Oquab, M.; Darcet, T.; Moutakanni, T.; Vo, H.; Szafraniec, M.; Khalidov, V.; Fernandez, P.; Haziza, D.; Massa, F. Dinov2: Learning robust visual features without supervision. *arXiv* **2023**, arXiv:2304.07193.
24. Zhang, J.; Herrmann, C.; Hur, J.; Polania Cabrera, L.; Jampani, V.; Sun, D.; Yang, M.H. A tale of two features: Stable diffusion complements dino for zero-shot semantic correspondence. In Proceeds of the 2023 Conference on Neural Information Processing Systems (NeurIPS 2023), New Orleans, LA, USA, 10–16 December 2023.
25. Yu, A.; Fridovich-Keil, S.; Tancik, M.; Chen, Q.; Recht, B.; Kanazawa, A. Plenoxels: Radiance fields without neural networks. *arXiv* **2021**, arXiv:2112.05131.
26. Lee, R.; Dudziak, Ł.; Abdelfattah, M.; Venieris, S.I.; Kim, H.; Wen, H.; Lane, N.D. *Journey Towards Tiny Perceptual Super-Resolution, European Conference on Computer Vision*; Springer International Publishing: Cham, Switzerland, 2020; pp. 85–102.
27. Kinoshita, Y.; Kiya, H. Fixed smooth convolutional layer for avoiding checkerboard artifacts in CNNs. In Proceedings of the ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Barcelona, Spain, 4–8 May 2020; IEEE: Piscataway, NJ, USA, 2020; pp. 3712–3716.
28. Yan, H.; Wang, Z.; Xu, Z.; Wang, Z.; Wu, Z.; Lyu, R. Research on Image Super-Resolution Reconstruction Mechanism based on Convolutional Neural Network. *arXiv* **2024**, arXiv:2407.13211.
29. Menis-Mastromichalakis, O.; Sofou, N.; Stamou, G. Deep ensemble art style recognition. In Proceedings of the 2020 International Joint Conference on Neural Networks (IJCNN), Glasgow, UK, 19–24 July 2020; IEEE: Piscataway, NJ, USA, 2020; pp. 1–8.
30. Wu, T.; Zhang, J.; Fu, X.; Wang, Y.; Ren, J.; Pan, L.; Wu, W.; Yang, L.; Wang, J.; Qian, C.; et al. Omniobject3d: Large-vocabulary 3d object dataset for realistic perception, reconstruction and generation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Vancouver, BC, Canada, 17–24 June 2023; pp. 803–814.
31. Li, H.; Li, Y.; Wang, H.; Zeng, J.; Xu, H.; Cai, P.; Chen, L.; Yan, J.; Xu, F.; Xiong, L.; et al. Open-sourced data ecosystem in autonomous driving: the present and future. *arXiv* **2023**, arXiv:2312.03408.
32. Severo, D.; Theis, L.; Ballé, J. The Unreasonable Effectiveness of Linear Prediction as a Perceptual Metric. *arXiv* **2023**, arXiv:2310.05986.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.