View-consistent Object Removal in Radiance Fields

Anonymous Authors

ABSTRACT

Radiance Fields (RFs) have emerged as a crucial technology for 3D scene representation, enabling the synthesis of novel views with remarkable realism. However, as RFs become more widely used, the need for effective editing techniques that maintain coherence across different perspectives becomes evident. Current methods primarily depend on per-frame 2D image inpainting, which often fails to maintain consistency across views, thus compromising the realism of edited RF scenes. In this work, we introduce a novel RF editing pipeline that significantly enhances consistency by requiring the inpainting of only a single reference image. This image is then projected across multiple views using a depth-based approach, effectively reducing the inconsistencies observed with per-frame inpainting. However, projections typically assume photometric consistency across views, which is often impractical in real-world settings. To accommodate realistic variations in lighting and viewpoint, our pipeline adjusts the appearance of the projected views by generating multiple directional variants of the inpainted image, thereby adapting to different photometric conditions. Additionally, we present an effective and robust multi-view object segmentation approach as a valuable byproduct of our pipeline. Extensive experiments demonstrate that our method significantly surpasses existing frameworks in maintaining content consistency across views and enhancing visual quality.

CCS CONCEPTS

• **Computing methodologies** → **Reconstruction**.

KEYWORDS

Visual editing, Image-based rendering, Radiance field, Multi-view consistency.

1 INTRODUCTION

Radiance Fields (RFs), such as Neural Radiance Fields (NeRF) [29] and 3D Gaussian Splatting (3D-GS) [17], are revolutionizing 3D scene representation and enhancing the realism of novel view synthesis. This technology holds great promise for Virtual and Augmented Reality (VR/AR), film production, and video game development. However, a significant challenge with the practical application of RFs is the difficulty of content modification, such as object removal. In implicit RF models (*e.g.*, NeRF), direct editing is challenging because scenes are encoded within neural network weights, which restricts precise user control over specific objects. In contrast, explicit RF models (*e.g.*, 3D-GS) encounter difficulties with unclear surface definitions, which hinder accurate object segmentation and complicate the editing process. Therefore, achieving high-quality modifications in RFs is nontrivial.

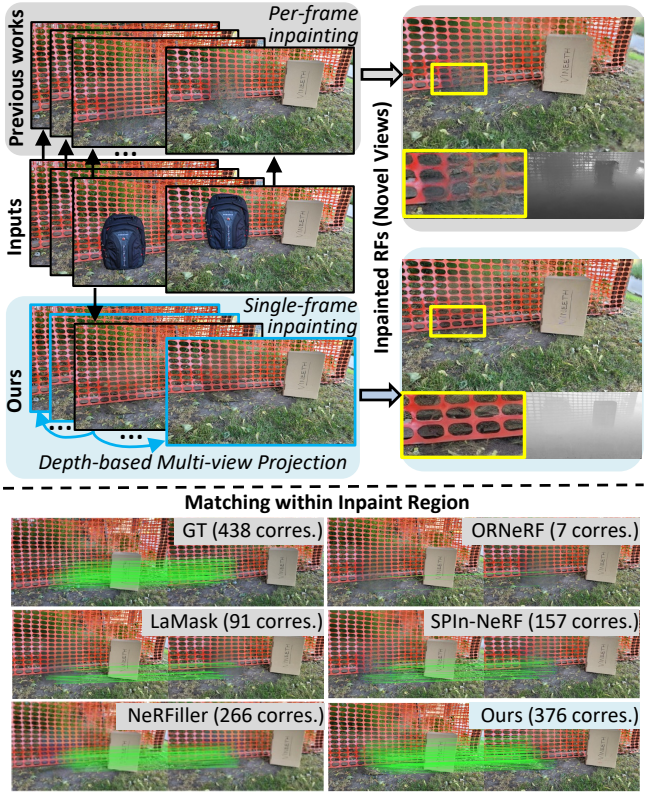

Figure 1: An illustration of our radiance field (RF) inpainting pipeline. Unlike conventional methods that inpaint on a per-frame basis, our approach inpaints a single reference image and applies depth-based projection to seamlessly extend the modifications across multiple views. We show that our method not only enhances the quality of inpainted RF scenes but also significantly improves correspondence between different perspectives.

3D scenes represented by RFs can be derived from sparse 2D images. To remove objects from these scenes, 2D inpainting methods are commonly used. Current works [30, 40, 59] typically begin with the creation of a multiview mask via image/video segmentation, which identifies the areas needing removal across different views. These specified areas are then independently inpainted for each view. However, these approaches have several shortcomings. The primary issue is the lack of consistency in object appearance and texture across different frames, as each frame is inpainted independently. This can lead to visual artifacts and unreliable scene

Unpublished working draft. Not for distribution.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
ACM MM, 2024, Melbourne, Australia
© 2024 Copyright held by the owner/author(s). Publication rights licensed to ACM.
ACM ISBN 978-x-xxxx-xxxx-x/YY/MM
https://doi.org/10.1145/nnnnnnn.nnnnnnn

geometry. Furthermore, achieving consistent segmentation itself is challenging with sparse inputs. Image-based segmentation methods can exhibit large variability between frames, while video-based segmentation methods struggle with images captured from infrequent or diverse angles. The limitations of these existing approaches highlight a significant gap in our ability to edit RF scenes without compromising their inherent realism and coherence.

In this paper, we proposed a novel RF inpainting method designed to maintain view consistency in object removal within 3D scenes (Fig. 1). This method simplifies the editing process by inpainting just a single, centrally-located reference image rather than multiple individual views. We then utilized depth-based projections to map the inpainted results from the reference view to other training views, effectively reducing inconsistencies commonly seen in per-frame inpainting and maintaining content consistency in the masked regions. Another key advantage of this method is the ability to utilize more advanced 2D inpainting techniques, such as diffusion-based generative models [11, 36]. These models produce highly realistic and detailed textures but typically falter in multi-view inpainting due to their stochastic nature. By applying these advanced techniques exclusively to the reference view, we can harness their strengths for high-quality inpainting while maintaining consistency across multiple views.

To effectively handle realistic variations in lighting and viewpoint, our pipeline strategically adjusts the appearance of projected views. Traditional depth-based projection methods transfer RGB values directly from the reference to the target regions under the assumption of uniform lighting conditions. However, this assumption often fails in real-world applications due to varying lighting and perspective shifts. To overcome this, we generate multiple directional variants of the inpainted reference image, each tailored to a different target direction. This is achieved by querying the reference view with color representations adjusted for each target direction. During the projection phase, we select the corresponding variant according to the target view, thus preserving both structural and view-dependent consistencies.

Another valuable byproduct of our pipeline is the depth-based multi-view segmentation method, which efficiently and robustly provides consistent masks across views. In summary, our proposed approach maintains consistency in both masks and inpaintings across all views, ensuring compatibility with various RF models, such as NeRF and 3D-GS. We have demonstrated the effectiveness of our method using these models, highlighting its versatility and potential to enhance RF scene editing capabilities.

The contributions of this paper are summarized as follows:

(1) A novel RF inpainting method that requires inpainting only one reference view, significantly enhancing efficiency and consistency across multiple views.
(2) A directional variants generation module adjusts the appearance of projected views to enhance the photorealism of the synthesized views.
(3) The development of a fast and robust multi-view segmentation approach to facilitate precise location and removal of objects across views.

2 RELATED WORK

2.1 Image Inpainting

Image inpainting is a problem that has been long studied in the field of computer vision [34]. Initial approaches to image inpainting primarily relied on the low-level features of damaged images, involving methods based on Partial Differential Equations (PDE) [1, 2, 44] and patch-based techniques [7, 10, 12]. Nowadays, deep learning based image inpainting methods has taken a dominate position. As mentioned by [34], deep learning based inpainting method can be classified as 1) deterministic image inpainting and 2) stochastic image inpainting. Given a image and its corresponding mask, deterministic image inpainting methods only produce an inpainting result, whereas stochastic image inpainting approaches are capable of generating several plausible outcomes through a process of random sampling.

As for deterministic methods, researchers often utilize three types of framework: single-shot, two-stage, and progressive methods. Single-shot methods [4, 25, 49, 53, 63] utilize an end-to-end generator network to output the inpainting result. Two-stage methods [37, 42, 55, 61, 62] consists of two generators and follows a coarse-to-fine strategy. The progressive methods [8, 22, 23, 66, 67] utilize multiple generators to inpaint the masked region in the given image in a iterative manner.

For stochastic methods, we can divided them into VAE-based methods [9, 18, 32, 45, 69, 70], GAN based methods [6, 16, 26, 68, 71], flow-based methods [5, 35, 48], MLM-based methods [46, 64] and Diffusion model-based methods. As diffusion model [11] has gained increasing popularity in recent years, latent diffusion models (LDMs) [24, 28, 54] has become the dominant method in the field of image inpainting.

In our work, we select diffusion model-based methods as they can produce more reasonable and photo-realistic inpainting results. Due to the nature of our work, we don't need to care about the stochastic property of diffusion models. While most of the previous works utilize LaMa [43], which is a deterministic method as they didn't explicitly handle the inconsistent inpainting issue.

2.2 3D Editing

With the emergence of NeRF and Gaussian Splatting, many excellent works [14, 15, 20, 21, 27, 31, 47, 57, 60, 65], have sprung up in the field of 3D scene editing. Some works [65] focus on the editing of the explicit geometry after training a NeRF. Peng et al. and Xu et al. [33, 56] try to make NeRF deformable and capable of animating general objects. Many works also put emphasis on object-centric editing. Wu et al. [52] proposed ObjectSDF, which is an object-compositional neural implicit representation, it is able to represent the surface of each object and the entire scene accurately. Yang et al. [57] proposed an object-compositional neural radiance field that is able to apply simple transformation and manipulation to the objects in the scene.

For object removal, there are also several works appear in recent years SPIn-NeRF [30], NeRF-In [40], Removing Objects From Neural Radiance Fields [51] and NeRFiller [50] demonstrate the ability to remove objects in NeRF. OR-NeRF [59] proposed a faster multi-view segmentation method and leverage TensoRF [3] to boost the rendering quality. Point'n Move [13] is able to handle object

removal in 3D Gaussian Splatting. All the object removal methods mentioned above are based on image editing, and then utilize inpainted images to train a inpainted radiance field. So the inconsistency of inpainting results in different views is a crucial problem to be solved. However, none of them have handled this issue perfectly. NeRF-In utilize pixel-wise MSE loss to simply supervise the content in the masked region, and does not have any further approach to deal with inconsistent inpainting result. SPIn-NeRF and OR-NeRF loosen the constrain provided by pixel-wise MSE loss and utilize perceptual loss to guide the optimization in the masked region to produce a more visually smooth result, but as viewpoint changes, the content in the inpainted region may still change slightly.

One recent work, NeRFiller [50], proposed to use Grid Prior (tile multiple images into a grid) for generating consistent inpainting images and propagate the inpainted part into the entire 3D scene in a iterative dataset update manner. According to our experiment, they did improve some 3D consistency, but the rendering quality is not satisfying. The reason for this is that they still need multiple times of inpainting. Though the consistency in maintained within each inpainting, there still exists inconsistency between different inpainting attempts.

In contrast, our method can maintain cross-view consistency during the inpainting process by explicitly projecting the generated content into all the training images. And due to the pre-mentioned drawback, the above approaches (except NeRFiller) cannot leverage advanced image inpainting methods e.g. Stable Diffusion to generate more photo-realistic results in complex environments. Because of the stochastic nature of diffusion model, even the input images are in the same environment, you can hardly get similar inpainting results. Since in our method we only need to inpaint one reference image, we do not have to deal with this issue.

3 PRELIMINARY: RADIANCE FIELDS

We demonstrate the effectiveness of our method using both implicit RF (*i.e.*, Neural Radiance Fields (NeRF) [29]) and explicit RF (*i.e.*, 3D Gaussian Splatting (3D-GS) [17]).

NeRF. NeRFs represent 3D scenes as Radiance Fields that maps the 3D coordinate x, y, z and the viewing direction θ, ϕ to color c and density σ. To get the color of a pixel, a ray will be shot through the pixel and then multiple points on the ray will be sampled. The color and density of each sampled point will be predicted by an MLP. Finally, volume rendering will be used to accumulate these sampled colors and render the pixel color $\widehat{C}$:

$$\widehat{C} = \sum_{i=1}^{N} T_i \left(1 - \exp\left(-\sigma_i \delta_i\right)\right) c_i,$$

where $T_i = \exp\left(-\sum_{j=1}^{i-1} \sigma_j \delta_j\right)$ is accumulated transmittance to the current sample point t_i, representing the probability that light travels from the camera to the point without hitting any other particles, and δ_i is the distance between adjacent sample points on the ray. c_i, σ_i correspond to the color and density at t_i. Reconstruction loss between ground truth color C and the predicted color $\widehat{C}$ is calculated to supervise the training process of NeRF.

3D-GS. 3D Gaussian Splatting utilizes a set of 3D ellipsoids to explicitly represent a scene. Each ellipsoid is modeled by an anisotropic 3D gaussian, which is parameterized by a center point x (mean of gaussian) and a covariance matrix $\sum$. The color of each gaussian is parametrized by spherical harmonics.

During the rendering process, 3D gaussians are first projected into image plane as 2D gaussians. Then the color of each pixel is calculated through the alpha-blending process over the points overlapping that pixel.

$$\widehat{C} = \sum_{i \in \mathcal{N}} c_i \alpha_i \prod_{j=1}^{i-1} \left(1 - \alpha_j\right),$$

where c_i is the color of each point calculated through spherical harmonics, and α_i is the opacity calculated from the covariance matrix $\sum'$. The rendered color is used to calculate the reconstruction loss with the ground truth color to optimize the 3D gaussians.

4 METHOD

In this part, we will describe our proposed method to maintain cross-view consistency for object removal in RFs and dive deeper into the details of each step in the following sections.

Our entire pipeline is shown in Fig. 2. We first select a camera with the least average distance on SO(3) manifold to all other cameras in the training data as the reference view. Then, the reference view is processed to get the mask M_r, inpainted reference view I_r, the depth map D_r of I_r (section 4.1). We then utilize depth-based projections to transfer the inpainted results from the reference view to other views, generating multi-view segmentation and inpainting results (section 4.2). Finally, an inpainted Radiance Field will be trained using the set of inpainted training images with the following reconstruction loss:

$$\mathcal{L}_{\text{rec}} = \sum_{k=0}^{N} \left\| \widehat{I}_k - I_k \right\|^2,$$

where I_k is the inpainted images via multi-view projection, $\widehat{I}_k$ is the generated results of inpainted RF, and N is the number of images.

4.1 High-quality Single-view Processing

We initiate our methodology by selecting a reference camera pose from the training dataset; this camera pose is identified as having the minimal average distance to all other poses on the SO(3) manifold. The processing of the chosen reference view involves three key steps: masking, inpainting, and depth estimation, yielding three outputs: the mask M_r, the inpainted image I_r, and the depth map D_r, respectively. These outputs are crucial for subsequent multi-view projection and inpainting tasks.

Mask Generation and Image Inpainting. To generate the mask M_r of the reference image, we employ the Segment Anything Model (SAM) [19], an advanced off-the-shelf model known for its efficiency and accuracy in image segmentation. For the inpainted image I_r of the reference view, we leverage a pretrained 2D inpainting model (*i.e.*, Stable Diffusion [36]) to fill the masked region with realistic texture and fine details.

Depth Map Estimation and Alignment. Generating the depth map D_r presents unique challenges, particularly regarding accuracy and smoothness. Previous approaches [30, 59] have relied on the trained RFs (*e.g.*, NeRF and 3D-GS) to derive depth information.

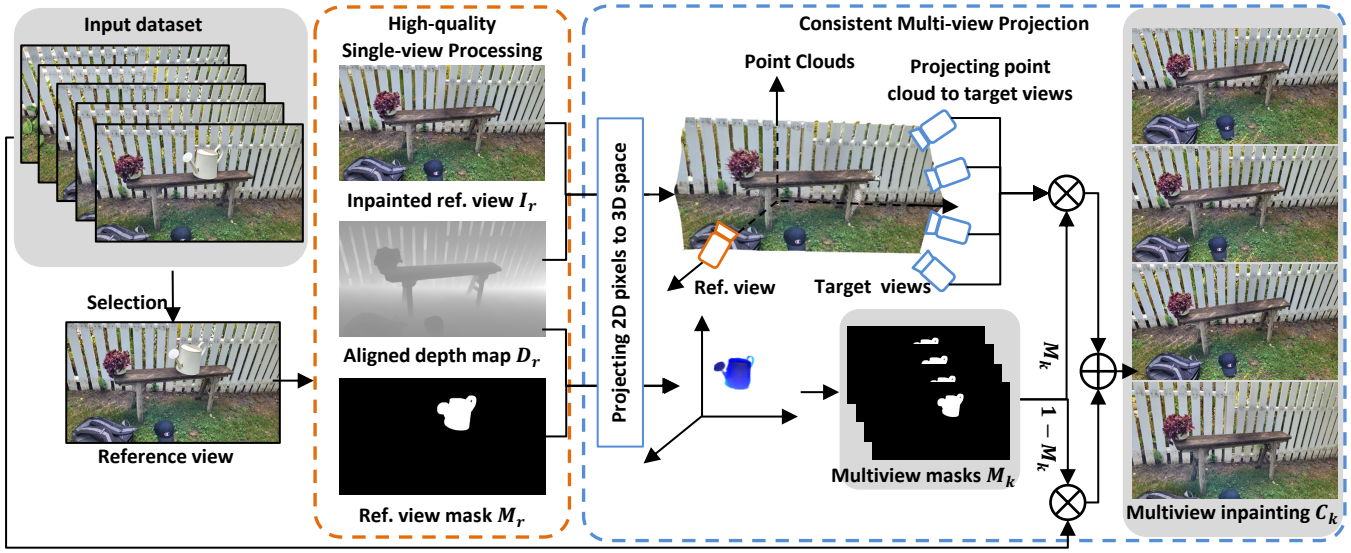

Figure 2: An overview of our method: we initiate our methodology by selecting a reference camera pose from the training dataset; this camera pose is identified as having the minimal average distance to all other poses on the SO(3) manifold. The processing of the chosen reference view involves three key steps: masking, inpainting, and depth estimation, yielding three outputs: the mask M_r, the inpainted image I_r, and the depth map D_r, respectively. These outputs are then used for multi-view projection, yielding a set of inpainted images from multiple views. Finally, an inpainted Radiance Field will be trained using these inpainted images.

However, RFs are sensitive to noise in the input data, which can degrade the depth map with artifacts and uneven surfaces. Such degradation will lead to irregular projection gaps during the multi-view projection process that rely on depth information.

To achieve precise and coherent depth information for projection, we start by estimating the depth with a monocular depth estimation method (*i.e.*, Depth-Anything [58]), producing an initial smooth depth map D_{init}. To resolve the scale ambiguity inherent in monocular depth estimation, we align D_{init} with sparse depth data D_{col} generated from the Structure-from-Motion (SfM) library COLMAP [39] to accurately scale the depth.

We approach the depth alignment between D_{col} and D_{init} as a least square problem, aiming to minimize the cost function:

$$\mathcal{L}_{\text{align}} = \sum_{i \in D_{col} \odot (1 - M_r)} D^i_{col} - (a \cdot D^i_{init} + b),$$

where D^i_{col} and D^i_{init} represent the depth of the i^{th} pixel in D_{col} and D_{init}, respectively, while a and b are the scale coefficients. To ensure accuracy around the object, we omit depth pixels both within the mask region M_r and those significantly distant from it. Once the optimal scale coefficients (*i.e.*, a^* and b^*) are determined, the final aligned depth D_r is calculated as:

$$D_r = a^* \cdot D_{init} + b^*.$$

This depth estimation and alignment strategy ensures our depth map D_r is not only accurate but also exhibits a smooth gradient, which is essential for error-free multi-view projection and inpainting workflows.

4.2 Multi-view Consistent Inpainting

Inpainting via Projection. Inspired by depth image-based rendering (DIBR) techniques, we utilize depth-based projections to transfer the inpainted results from the reference view to other views, after processing the single reference view. This approach addresses the common inconsistencies found in per-frame inpainting and ensures content consistency within the masked regions.

Upon obtaining the inpainted reference image I_r and its corresponding depth map D_r, our goal is to project I_r onto a target view k, generating the inpainted image I_k. We start by backprojecting each 2D pixel in I_r into 3D space to create a point cloud $\mathbf{c}_r$ using its depth information. Specifically, the i^{th} pixel in I_r, denoted as I^i_r, corresponds to a point $\mathbf{c}^i_r$ in 3D space. The coordinates of each point $\mathbf{c}^i_r$ are calculated as follows:

$$\mathbf{c}^i_r = \begin{bmatrix} X^i_r \\ Y^i_r \\ Z^i_r \end{bmatrix} = D_r(u^i_r, v^i_r) \cdot K^{-1} \cdot \begin{bmatrix} u^i_r \\ v^i_r \\ 1 \end{bmatrix},$$

where K is the camera intrinsic matrix and (u^i_r, v^i_r) are the coordinates of the i^{th} pixel in inpainted reference image I_r. $D_r(x, y)$ represents the depth value at position (x, y) in the depth map D_r.

Following this, we project the point cloud $\mathbf{c}_r$ to the new viewpoint k through a relative transformation matrix T between the reference and the target viewpoints:

$$c^i_k = \mathbf{T} \cdot c^i_r,$$

where c_k^i is the projected point in view k. The coordinates of points in the target image space are then calculated as:

$$\begin{bmatrix} u_k^i \\ v_k^i \\ 1 \end{bmatrix} = K \cdot \frac{1}{Z_k^i} \cdot c_k^i,$$

where Z_k^i is the depth of point c_k^i.

The pixel values in the target view's masked region are then replaced by the corresponding projected pixel values:

$$I_k(u_k^i, v_k^i) = I_r(u_r^i, v_r^i).$$

After projection, another crucial task is to handle the projection gaps due to occlusions. We utilize LaMa [43] to inpaint these small and regular gaps, resulting in a set of refined projection results $\{I_k\}$, $k = 0, 1, 2, \cdots, N-1$, where I_k represents the projected inpainting result from I_r to view k.

Similarly, given the mask M_r of the reference view along with the depth information D_r, we can project M_r onto target view k and get the corresponding mask M_k. This method enables the automatic generation of robust and view-consistent segmentations across multiple views.

View Dependent Effect. Directly projecting and propagating the inpainting result from the reference view to all the other training views has an assumption of photometric consistency. However, this assumption often fails in real-world scenarios, where lighting and viewpoint variations are common. To address these challenges and better adapt to realistic variation, we strategically adjust the appearance of projected views to better match their target settings. The approach involves generating multiple directional variants of the inpainted reference image, each tailored to a specific target direction. During the projection phase, we select and utilize the variant that best corresponds to the target view.

To generate these directional variants, we first train a Radiance Field with the original training set before inpainting. Then, we extract the view-dependent appearance encoded in the trained RF representation. This is done by maintaining the camera's viewpoint as fixed at the reference view while varying the queried viewing directions with the target views. This process results in N-1 directional variants of the reference image, each reflecting different lighting conditions. We then utilize Stable Diffusion to inpaint these reference views. Empirical evidence suggests that the content generated by Stable Diffusion maintains geometric consistency under minor variations in lighting conditions. By leveraging this property, we are able to accurately generate and project these variants across different views, ensuring that the adjustments align well with the varying conditions of each target view.

Depth-Based Occlusion Correction. During the projection process, multiple points from the reference point cloud $\mathbf{c}_r$ may be projected onto the same pixel in the target view. Thus, we need to maintain a Z-buffer to ensure that the points with small depth values will remain on the image plane.

Besides, some pixels primarily occluded may be revealed at the surface accidentally. The reason why this happens is that the inpainted reference view may have some content that should be occluded in the target view. They are now exposed at the surface because the foreground content that should cover them is not available in the reference view, which means they are in the projection gap and thus not available.

To deal with this issue we introduce depth prior during the projection process. Briefly speaking, we first estimate the depth of each target view as described in section 4.1. Then during the projection process, we utilize the estimated depth map as a depth prior, and reject any pixel that has a depth larger than the corresponding depth prior. The detailed algorithm to deal with occlusion and de-occlusion issue is shown in algorithm 1 as pseudo-code.

Algorithm 1 This pseudo-code describes how to utilize Z-buffer and Depth Prior to help deal with the occlusion and de-occlusion issue during depth-based projection process.

Require: $\mathbf{c}_r$, $\mathbf{D}_r$, $\mathbf{T}$, $\mathbf{D}_{prior}$
  $z\_buf[1 \dots n] \leftarrow$ new Array(n)
  for $i = 1$ to n **do**
    $z\_buf[i] \leftarrow \infty$
  end for
  for $k = 1$ to n **do**
    $u_k^i, v_k^i, Z_k^i \leftarrow \mathbf{DIBR}(\mathbf{c}_r, \mathbf{D}_r, \mathbf{T})$
    if $0 \le u_k^i < w$ **and** $0 \le u_k^i < h$ **and** $Z_k^i < z\_buf[i]$ **then**
      if $\mathbf{D}_{prior}(u_k^i, v_k^i) - Z_k^i > \epsilon$ **then**
        $I_k(u_k^i, v_k^i) \leftarrow I_r(u_r^i, v_r^i)$
        $z\_buf[i] \leftarrow Z_k^i$
      end if
    end if
  end for

5 EXPERIMENTS

5.1 Dataset and Implementation Details

All the following experiments are accomplished based on the SPIn-NeRF dataset [30], which was designed specifically for 3D inpainting. SPIn-NeRF dataset contains 10 scenes, including both indoor and outdoor scenarios. Within each scene, there are 60 training images including the unwanted object, and 40 ground truth images with the unwanted object removed. The dataset also provide human annotated segmentation masks for each training images.

We run both vanilla NeRF and 3D-GS based on our inpainting results, without additional modification on loss function or training procedure to show the effectiveness of our proposed method.

For the initialization of 3D-GS, we first remove the unnecessary points in the masked area from the sparse point cloud generated by colmap, and then leverage the aligned depth estimation results produced in section 4.1 to serve as the initialization for the mean of 3D Gaussians inside the masked region.

5.2 Radiance Field Inpainting

For the quantatitive comparison on Radiance Field Inpainting, we report the average peak signal-to-noise ratio (PSNR), the average learned perceptual image patch similarity (LPIPS), and the average Fréchet inception distance (FID) between the rendered test view and the ground truth test image provided by the SPIn-NeRF dataset.

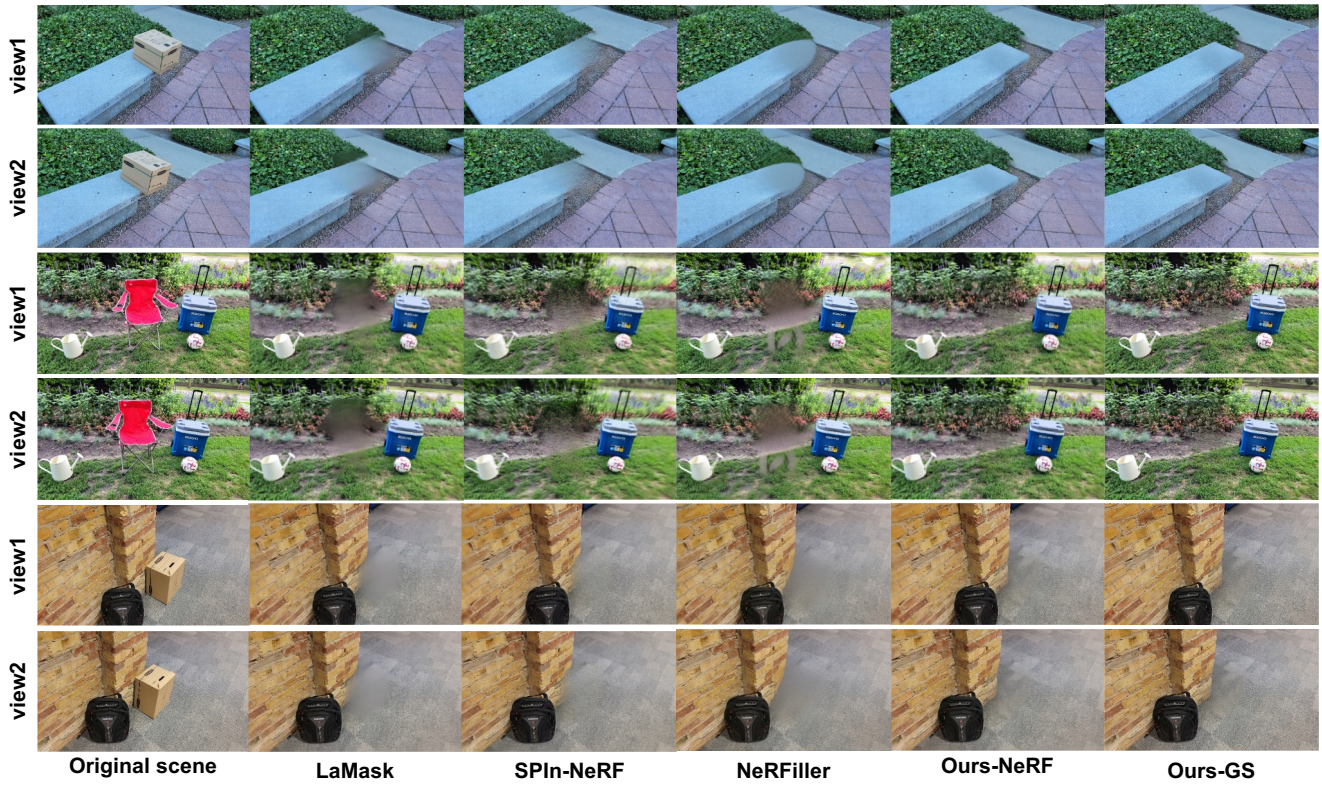

Figure 3: Qualitative comparison between our methods and baseline methods. For each scene, we show images from two different views to compare both rendering quality and cross-view consistency.

Note that the ground truth images are only used for evaluation, and are not required during training. Our baselines are the following:

(1) **LaMask** - Inpainting all the training images with LaMa [43], and train a vanilla NeRF without any other techniques based on these inpainted images.
(2) **SPIn-NeRF** [30] - Based on LaMask, utilize depth inpainting as depth supervision and apply perceptual loss, LPIPS within the mask region to solve the blurry issue caused by inconsistent inpainting.
(3) **OR-NeRF (TensoRF)** [59] - Enhanced version of SPIn-NeRF, using TensoRF instead of vanilla NeRF.
(4) **NeRFiller** [50] - NeRFiller utilize grid prior (tile the input images into a grid and treat the entire grid as a single inpainting target) to generate more consistent inpaintings. And proposed an iterative 3D scene optimization method to maintain global 3D consistency.

The quantatitive results are shown in Table 1. Our inpainting method trained with Gaussian Splatting (Ours-GS) achieves the best performance in terms of LPIPS and FID score, and Ours-NeRF outperforms all the other models in PSNR. It is worth mentioning that though Ours-NeRF utilizes vanilla NeRF as backend, it still achieve competitive or even better results compared with ORNeRF (TensoRF backend) and NeRFiller (Nerfacto backend). We also show some qualitative comparison in Fig. 3.

Methods	PSNR ↑	LPIPS ↓	FID ↓
SPIn-NeRF	20.63	0.39	68.23
LaMask	20.27	0.41	63.06
ORNeRF-TensoRF	18.53	0.25	48.28
NeRFiller	19.71	0.37	72.79
Ours-GS	20.22	**0.21**	**35.69**
Ours-NeRF	**20.82**	0.38	47.79

Table 1: Quantitative comparison of our inpainting method with ground truth object masks

5.3 Multi-view Consistency

Inpainting Consistency. In this section, we evaluate the multi-view consistency of our methods against the baseline approaches. We apply widely used off-the-shelf image feature matching methods LoFTR [41] and SuperGlue [38] to check the number of correspondence between the image pairs rendered by ours and baseline methods. The comparison results are shown in Table 2. For both feature matching methods, we randomly sample 100 images pairs to calculate the correspondence and only the matchings within the masked region are taken into consideration. For LoFTR, we only calculate the correspondence with confidence level higher that 0.95. We use pretrained weight "indoor" for scene 9, book and trash

Methods	LoFTR	SuperGlue
SPIn-NeRF	154.03	19.44
LaMask	105.79	23.11
ORNeRF-TensoRF	34.48	18.07
NeRFiller	201.34	22.94
Ours-GS	283.52	40.25
Ours-NeRF	**319.04**	**64.40**

Table 2: Number of correspondence found between pairs of rendered images. A higher correspondence value indicates better geometry consistency.

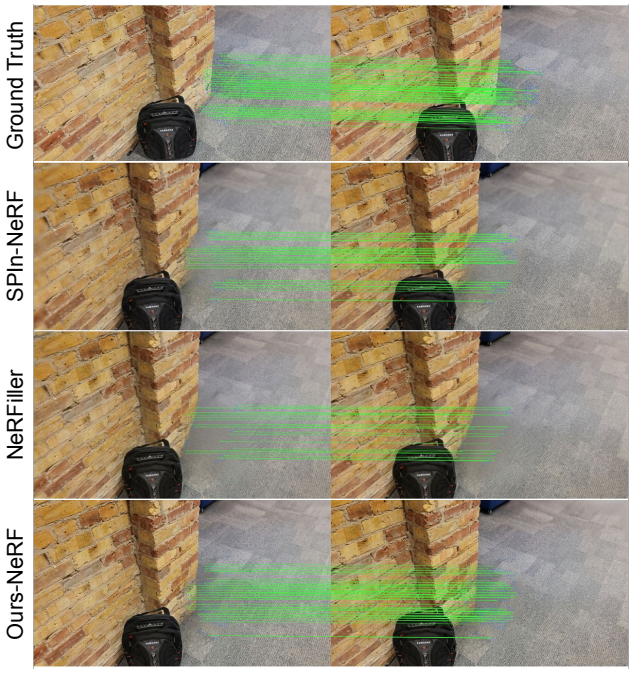

Figure 4: Visualization of feature matching results within the masked region. Ground Truth, SPIn-NeRF, NeRFiller, and Ours-NeRF have number of matchings 329, 193 , 84 and 324 respectively. The original scene picture is shown in Fig. 3

and all the other scenes are evaluated with "outdoor" weight. As for quantitative results, our inpainting approached outperform the baseline methods in both of the matching method.

We also visualize the matching results of LoFTR in Fig. 4 for comparison. The first row in Fig. 4 shows the matching result between two ground truth images with unwanted objects removed provided by the SPIn-NeRF dataset.

Mask Consistency. The mask consistency across different views is also quite crucial in the Radiance Field editing process. Inconsistent masks will cause inconsistent inpainting and thus break the 3D consistency. Here, we compare our method with two segmentation methods proposed by OR-NeRF to demonstrate our mask consistency. OR-NeRF proposed two segmentation methods 1) point prompt based and 2) text prompt based. The point prompt based

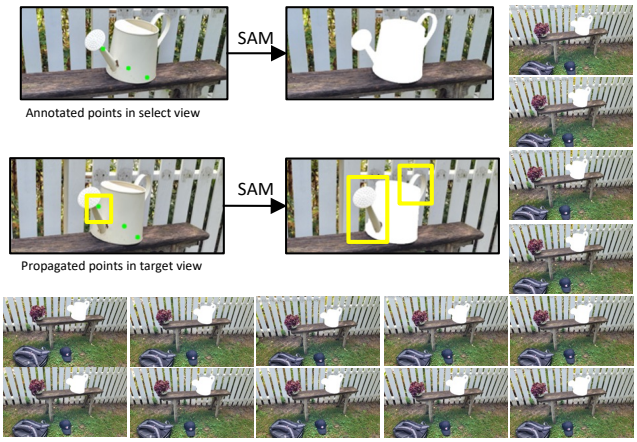

Figure 5: Failure case of OR-NeRF (point prompt) is on the upper left corner. The first row shows the manually annotated point prompts in a selected view and its corresponding mask generated by SAM. The second row shows the propagated point prompts to another view and its corresponding mask. We can see that one of the propagated point prompt does not lay on the expected region and thus the generated mask is not completed.

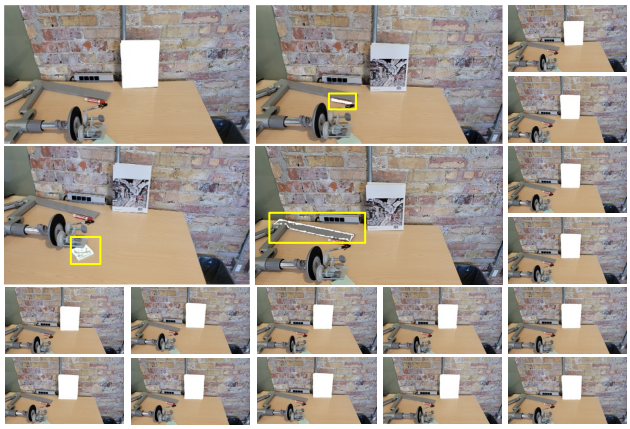

Figure 6: Failure case of OR-NeRF (text prompt) is on the upper left corner. We use the text prompt "book" to do the segmentation. We can see that SAM may incorrectly segment the pen, sticky note and the metal bar on the table as "book".

one requires manually annotating some points on a selected 2D image and utilize the sparse point cloud generated by colmap to spread the point prompt to all the other views. The text prompt one uses a single text prompt for SAM to just the segmentation result for all the images. However, both of them have some drawbacks. For point prompt, not all the annotated points can be found in the point cloud, and thus they need to find a closest point as replacement, which may cause an offest during propagation. For text prompt, it is quite hard to find a universal prompt that works for all the images.

Method	Acc. ↑	IoU ↑	Dice ↑
SPIn-NeRF	98.91	91.66	-
OR-NeRF (text)	97.78	72.75	84.26
OR-NeRF (points)	**99.63**	94.07	96.84
Ours	99.48	**94.27**	**96.98**

Table 3: quantitative comparison between our proposed multi-view segmentation methods and the baseline methods

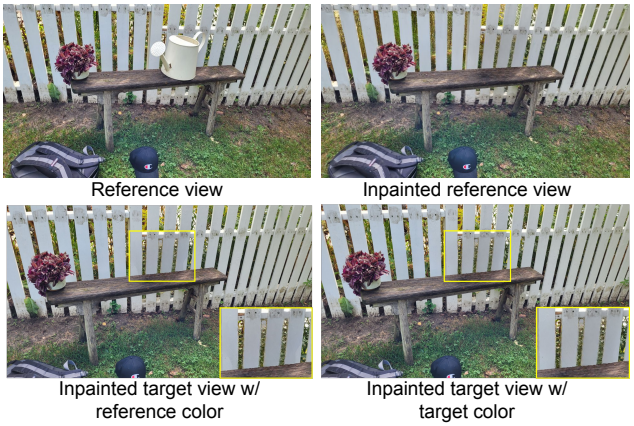

Figure 7: Ablation study for view-dependent effect. reference view of different lighting conditions. The first row shows the selected reference view and its corresponding inpainting result. The second row shows the inpainting result w/ and wo/ using the lighting variant reference view.

We show some failure cases of OR-NeRF and also our segmentation results over the same scene in Fig. 5 and Fig. 6 to proof the mask consistency of our proposed method. Our results are shown at the periphery of these two figures.

We then quantitatively compared our depth projection based multi-view segmentation method with the MVSeg model provided by SPIn-NeRF and the points/text prompt based multi-view segmentation method propsed by OR-NeRF. We report average accuracy, intersection over union (IoU) and Dice score between the human-annotated ground truth mask and the mask predicted by different approaches, the numerical results are shown in Table 3. For SPIn-NeRF, as they didn't report Dice score in their paper and the code for MVSeg is currently not available, we just leave it blank.

5.4 Ablation Study

View-dependent Effect. We rendered multiple directional variants of reference views as indicated in section 4.2. In this ablation study, we show the effectiveness of this module. Fig. 7 shows a comparison between the inpainting result of the target view projected by the original reference view and the result projected by the reference view with lighting variation. We can see that if we directly project the inpainted area to the target view without changing the lighting condition, it will result in an obvious contour around the inpainting region. After applying the target color to the reference view, this phenomenon is eliminated.

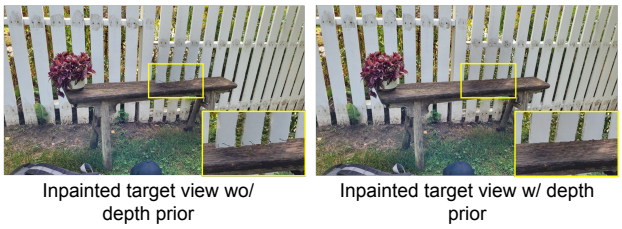

Figure 8: Ablation study for depth-based occlusion correction.

Methods	PSNR ↑	LPIPS ↓	FID ↓
Ours-GS (GT mask)	20.22	0.21	35.69
Ours-GS (Our mask)	20.14	0.21	35.32
Ours-NeRF (GT mask)	20.82	0.38	47.79
Ours-NeRF (Our mask)	20.68	0.39	48.15

Table 4: Quantitative comparison between the Radiance Field inpainting results using human-annotated mask and the mask generated by our proposed segmentation methods.

Depth-Based Occlusion Correction. As claimed in section 4.2, z-buffer and depth prior are used to solve the issue of occlusion and de-occlusion of projected points. Here we visualize the above mentioned issue and show the improved inpainting result with depth-based occlusion correction. From the left image in Fig. 8, we can see that some part of the barrier ought to be occluded by the bench is now revealed at the surface. After applying the depth-based occlusion correction, the occlusion relationship between the bench and the barrier is corrected.

Multi-view Segmentation. We also quantitatively compare the RF Inpainting results using masks generated with our segmentation method and the ground truth masks provided by the dataset (Table 4). It shows that using the masks generated by our method only results in subtle performance degradation in RF inpainting.

6 CONCLUSION

Our work introduces a novel RF editing pipeline designed to overcome the 3D inconsistency issue during 3D object removal. By employing a strategy of inpainting a single reference image followed by depth-based projection, our method efficiently extends the inpainted effects across multiple views, thereby minimizing the inconsistencies observed with per-frame inpainting approaches. Furthermore, we also accommodate view-dependent effect by adjusting observed colors based on the viewing direction, which is determined during the color querying phase. Through rigorous testing, we demonstrate that our method maintains content rationality and significantly improves the visual quality of RF scenes, which marks a substantial advancement over existing frameworks.

REFERENCES

[1] C. Ballester, M. Bertalmio, V. Caselles, G. Sapiro, and J. Verdera. 2001. Filling-in by joint interpolation of vector fields and gray levels. *IEEE Transactions on Image Processing* 10, 8 (2001), 1200–1211. https://doi.org/10.1109/83.935036

[2] Marcelo Bertalmio, Guillermo Sapiro, Vincent Caselles, and Coloma Ballester. 2000. Image inpainting. In *Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH '00)*. ACM Press/Addison-Wesley Publishing Co., USA, 417–424. https://doi.org/10.1145/344779.344972

[3] Anpei Chen, Zexiang Xu, Andreas Geiger, Jingyi Yu, and Hao Su. 2022. TensoRF: Tensorial Radiance Fields. In *European Conference on Computer Vision (ECCV)*.

[4] Ye Deng, Siqi Hui, Sanping Zhou, Deyu Meng, and Jinjun Wang. 2021. Learning contextual transformer network for image inpainting. In *Proceedings of the 29th ACM international conference on multimedia*. 2529–2538.

[5] Laurent Dinh, David Krueger, and Yoshua Bengio. 2014. Nice: Non-linear independent components estimation. *arXiv preprint arXiv:1410.8516* (2014).

[6] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2014. Generative adversarial nets. *Advances in neural information processing systems* 27 (2014).

[7] Qiang Guo, Shanshan Gao, Xiaofeng Zhang, Yilong Yin, and Cai ming Zhang. 2018. Patch-Based Image Inpainting via Two-Stage Low Rank Approximation. *IEEE Transactions on Visualization and Computer Graphics* 24 (2018), 2023–2036. https://api.semanticscholar.org/CorpusID:8348376

[8] Zongyu Guo, Zhibo Chen, Tao Yu, Jiale Chen, and Sen Liu. 2019. Progressive Image Inpainting with Full-Resolution Residual Network. *Proceedings of the 27th ACM International Conference on Multimedia* (2019). https://api.semanticscholar.org/CorpusID:198229717

[9] Xintong Han, Zuxuan Wu, Weilin Huang, Matthew R Scott, and Larry S Davis. 2019. Finet: Compatible and diverse fashion image inpainting. In *Proceedings of the IEEE/CVF international conference on computer vision*. 4481–4491.

[10] Jan Herling and Wolfgang Broll. 2014. High-Quality Real-Time Video Inpaintingwith PixMix. *IEEE Transactions on Visualization and Computer Graphics* 20, 6 (2014), 866–879. https://doi.org/10.1109/TVCG.2014.2298016

[11] Jonathan Ho, Ajay Jain, and Pieter Abbeel. 2020. Denoising Diffusion Probabilistic Models. *arXiv preprint arxiv:2006.11239* (2020).

[12] Jia Bin Huang, Sing Bing Kang, Narendra Ahuja, and Johannes Kopf. 2014. Image completion using planar structure guidance. *ACM Transactions on Graphics* 33, 4 (2014). https://doi.org/10.1145/2601097.2601205 Funding Information: We thank the flickr users who put their images under Creative Commons license or allowed us to use them. For a detailed list of contributors to our image dataset, please refer to the accompanying project website. The support of the Office of Naval Research under grant N00014-12-1-0259 is gratefully acknowledged.; 41st International Conference and Exhibition on Computer Graphics and Interactive Techniques, ACM SIGGRAPH 2014 ; Conference date: 10-08-2014 Through 14-08-2014.

[13] Jiajun Huang and Hongchuan Yu. 2023. Point'n Move: Interactive Scene Object Manipulation on Gaussian Splatting Radiance Fields. *arXiv preprint arXiv:2311.16737* (2023).

[14] Ru-Fen Jheng, Tsung-Han Wu, Jia-Fong Yeh, and Winston H Hsu. 2022. Free-form 3D scene inpainting with dual-stream GAN. *arXiv preprint arXiv:2212.08464* (2022).

[15] Kacper Kania, Kwang Moo Yi, Marek Kowalski, Tomasz Trzciński, and Andrea Tagliasacchi. 2022. CoNeRF: Controllable Neural Radiance Fields. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*.

[16] Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. 2020. Analyzing and improving the image quality of stylegan. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 8110–8119.

[17] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 2023. 3D Gaussian Splatting for Real-Time Radiance Field Rendering. *ACM Transactions on Graphics* 42, 4 (July 2023). https://repo-sam.inria.fr/fungraph/3d-gaussian-splatting/

[18] Diederik P Kingma and Max Welling. 2014. Auto-encoding variational {Bayes}. In *Int. Conf. on Learning Representations*.

[19] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. 2023. Segment anything. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 4015–4026.

[20] Zhengfei Kuang, Fujun Luan, Sai Bi, Zhixin Shu, Gordon Wetzstein, and Kalyan Sunkavalli. 2022. PaletteNeRF: Palette-based Appearance Editing of Neural Radiance Fields. *arXiv preprint arXiv:2212.10699* (2022).

[21] Verica Lazova, Vladimir Guzov, Kyle Olszewski, Sergey Tulyakov, and Gerard Pons-Moll. 2022. Control-NeRF: Editable Feature Volumes for Scene Rendering and Manipulation. *arXiv preprint arXiv:2204.10850* (2022).

[22] Jingyuan Li, Fengxiang He, Lefei Zhang, Bo Du, and Dacheng Tao. 2019. Progressive reconstruction of visual structure for image inpainting. In *Proceedings of the IEEE/CVF international conference on computer vision*. 5962–5971.

[23] Jingyuan Li, Ning Wang, Lefei Zhang, Bo Du, and Dacheng Tao. 2020. Recurrent feature reasoning for image inpainting. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 7760–7768.

[24] Wenbo Li, Xin Yu, Kun Zhou, Yibing Song, Zhe Lin, and Jiaya Jia. 2023. Image Inpainting via Iteratively Decoupled Probabilistic Modeling. arXiv:2212.02963 [cs.CV]

[25] Guilin Liu, Fitsum A. Reda, Kevin J. Shih, Ting-Chun Wang, Andrew Tao, and Bryan Catanzaro. 2018. Image Inpainting for Irregular Holes Using Partial Convolutions. In *The European Conference on Computer Vision (ECCV)*.

[26] Hongyu Liu, Ziyu Wan, Wei Huang, Yibing Song, Xintong Han, and Jing Liao. 2021. Pd-gan: Probabilistic diverse gan for image inpainting. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 9371–9381.

[27] Steven Liu, Xiuming Zhang, Zhoutong Zhang, Richard Zhang, Jun-Yan Zhu, and Bryan Russell. 2021. Editing Conditional Radiance Fields. In *Proceedings of the International Conference on Computer Vision (ICCV)*.

[28] Andreas Lugmayr, Martin Danelljan, Andrés Romero, Fisher Yu, Radu Timofte, and Luc Van Gool. 2022. RePaint: Inpainting using Denoising Diffusion Probabilistic Models. *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2022), 11451–11461. https://api.semanticscholar.org/CorpusID:246240274

[29] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. 2020. NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis. In *European Conference on Computer Vision*. Springer, 405–421.

[30] Ashkan Mirzaei, Tristan Aumentado-Armstrong, Konstantinos G. Derpanis, Jonathan Kelly, Marcus A. Brubaker, Igor Gilitschenski, and Alex Levinshtein. 2023. SPIn-NeRF: Multiview Segmentation and Perceptual Inpainting with Neural Radiance Fields. In *CVPR*.

[31] Ashkan Mirzaei, Yash Kant, Jonathan Kelly, and Igor Gilitschenski. 2022. Laterf: Label and text driven object radiance fields. In *European Conference on Computer Vision*. Springer, 20–36.

[32] Jialun Peng, Dong Liu, Songcen Xu, and Houqiang Li. 2021. Generating diverse structure for image inpainting with hierarchical VQ-VAE. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 10775–10784.

[33] Yicong Peng, Yichao Yan, Shengqi Liu, Yuhao Cheng, Shanyan Guan, Bowen Pan, Guangtao Zhai, and Xiaokang Yang. 2022. CageNeRF: Cage-based Neural Radiance Field for Generalized 3D Deformation and Animation. In *Advances in Neural Information Processing Systems*, S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh (Eds.), Vol. 35. Curran Associates, Inc., 31402–31415. https://proceedings.neurips.cc/paper_files/paper/2022/file/cb78e6b5246b03e0b82b4acc8b11cc21-Paper-Conference.pdf

[34] Weize Quan, Jiaxi Chen, Yanli Liu, Dong-Ming Yan, and Peter Wonka. 2024. Deep Learning-Based Image and Video Inpainting: A Survey. *International Journal of Computer Vision* (2024), 1–34.

[35] Danilo Rezende and Shakir Mohamed. 2015. Variational inference with normalizing flows. In *International conference on machine learning*. PMLR, 1530–1538.

[36] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. 2021. High-Resolution Image Synthesis with Latent Diffusion Models. arXiv:2112.10752 [cs.CV]

[37] Min-cheol Sagong, Yong-goo Shin, Seung-wook Kim, Seung Park, and Sung-jea Ko. 2019. Pepsi: Fast image inpainting with parallel decoding network. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 11360–11368.

[38] Paul-Edouard Sarlin, Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. 2020. SuperGlue: Learning Feature Matching with Graph Neural Networks. In *CVPR*. https://arxiv.org/abs/1911.11763

[39] Johannes L Schonberger and Jan-Michael Frahm. 2016. Structure-from-motion revisited. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 4104–4113.

[40] I-Chao Shen, Hao-Kang Liu, and Bing-Yu Chen. 2024. NeRF-In: Free-Form NeRF Inpainting with RGB-D Priors. *Computer Graphics and Applications (CG&A)* (2024).

[41] Jiaming Sun, Zehong Shen, Yuang Wang, Hujun Bao, and Xiaowei Zhou. 2021. LoFTR: Detector-Free Local Feature Matching with Transformers. *CVPR* (2021).

[42] Qianru Sun, Liqian Ma, Seong Joon Oh, Luc Van Gool, Bernt Schiele, and Mario Fritz. 2018. Natural and effective obfuscation by head inpainting. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 5050–5059.

[43] Roman Suvorov, Elizaveta Logacheva, Anton Mashikhin, Anastasia Remizova, Arsenii Ashukha, Aleksei Silvestrov, Naejin Kong, Harshith Goka, Kiwoong Park, and Victor Lempitsky. 2021. Resolution-robust Large Mask Inpainting with Fourier Convolutions. *arXiv preprint arXiv:2109.07161* (2021).

[44] David Tschumperlé and R. Deriche. 2005. Deriche, R.: Vector-valued image regularization with PDEs: a common framework for different applications. IEEE Trans. Pattern Anal. Machine Intell. 27, 506-517. *IEEE transactions on pattern analysis and machine intelligence* 27 (05 2005), 506–17. https://doi.org/10.1109/TPAMI.2005.87

[45] Ching-Ting Tu and Yi-Fu Chen. 2019. Facial image inpainting with variational autoencoder. In *2019 2nd international conference of intelligent robotic and control engineering (IRCE)*. IEEE, 119–122.

[46] Ziyu Wan, Jingbo Zhang, Dongdong Chen, and Jing Liao. 2021. High-fidelity pluralistic image completion with transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 4692–4701.

[47] Can Wang, Menglei Chai, Mingming He, Dongdong Chen, and Jing Liao. 2021. CLIP-NeRF: Text-and-Image Driven Manipulation of Neural Radiance Fields. *arXiv preprint arXiv:2112.05139* (2021).

[48] Cairong Wang, Yiming Zhu, and Chun Yuan. 2022. Diverse image inpainting with normalizing flow. In *European conference on computer vision*. Springer, 53–69.

[49] Wentao Wang, Jianfu Zhang, Li Niu, Haoyu Ling, Xue Yang, and Liqing Zhang. 2021. Parallel multi-resolution fusion network for image inpainting. In *Proceedings of the IEEE/CVF international conference on computer vision*. 14559–14568.

[50] Ethan Weber, Aleksander Holynski, Varun Jampani, Saurabh Saxena, Noah Snavely, Abhishek Kar, and Angjoo Kanazawa. 2024. NeRFiller: Completing Scenes via Generative 3D Inpainting. In *CVPR*.

[51] Silvan Weder, Guillermo Garcia-Hernando, Áron Monszpart, Marc Pollefeys, Gabriel Brostow, Michael Firman, and Sara Vicente. 2023. Removing Objects From Neural Radiance Fields. In *CVPR*.

[52] Qianyi Wu, Xian Liu, Yuedong Chen, Kejie Li, Chuanxia Zheng, Jianfei Cai, and Jianmin Zheng. 2022. Object-compositional neural implicit surfaces. In *European Conference on Computer Vision*. Springer, 197–213.

[53] Chaohao Xie, Shaohui Liu, Chao Li, Ming-Ming Cheng, Wangmeng Zuo, Xiao Liu, Shilei Wen, and Errui Ding. 2019. Image inpainting with learnable bidirectional attention maps. In *Proceedings of the IEEE/CVF international conference on computer vision*. 8858–8867.

[54] Shaoan Xie, Zhifei Zhang, Zhe Lin, Tobias Hinz, and Kun Zhang. 2023. Smartbrush: Text and shape guided object inpainting with diffusion model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 22428–22437.

[55] Wei Xiong, Jiahui Yu, Zhe Lin, Jimei Yang, Xin Lu, Connelly Barnes, and Jiebo Luo. 2019. Foreground-aware image inpainting. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 5840–5848.

[56] Tianhan Xu and Tatsuya Harada. 2022. Deforming Radiance Fields with Cages. In *ECCV*.

[57] Bangbang Yang, Yinda Zhang, Yinghao Xu, Yijin Li, Han Zhou, Hujun Bao, Guofeng Zhang, and Zhaopeng Cui. 2021. Learning Object-Compositional Neural Radiance Field for Editable Scene Rendering. In *International Conference on Computer Vision (ICCV)*.

[58] Lihe Yang, Bingyi Kang, Zilong Huang, Xiaogang Xu, Jiashi Feng, and Hengshuang Zhao. 2024. Depth Anything: Unleashing the Power of Large-Scale Unlabeled Data. In *CVPR*.

[59] Youtan Yin, Zhoujie Fu, Fan Yang, and Guosheng Lin. 2023. OR-NeRF: Object Removing from 3D Scenes Guided by Multiview Segmentation with Neural Radiance Fields. arXiv:2305.10503 [cs.CV]

[60] Hong-Xing Yu, Leonidas J. Guibas, and Jiajun Wu. 2022. Unsupervised Discovery of Object Radiance Fields. In *International Conference on Learning Representations*.

[61] Jiahui Yu, Zhe Lin, Jimei Yang, Xiaohui Shen, Xin Lu, and Thomas S Huang. 2018. Generative image inpainting with contextual attention. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 5505–5514.

[62] Jiahui Yu, Zhe Lin, Jimei Yang, Xiaohui Shen, Xin Lu, and Thomas S Huang. 2019. Free-form image inpainting with gated convolution. In *Proceedings of the IEEE/CVF international conference on computer vision*. 4471–4480.

[63] Tao Yu, Zongyu Guo, Xin Jin, Shilin Wu, Zhibo Chen, Weiping Li, Zhizheng Zhang, and Sen Liu. 2020. Region normalization for image inpainting. In *Proceedings of the AAAI conference on artificial intelligence*. 12733–12740.

[64] Yingchen Yu, Fangneng Zhan, Rongliang Wu, Jianxiong Pan, Kaiwen Cui, Shijian Lu, Feiying Ma, Xuansong Xie, and Chunyan Miao. 2021. Diverse image inpainting with bidirectional and autoregressive transformers. In *Proceedings of the 29th ACM International Conference on Multimedia*. 69–78.

[65] Yu-Jie Yuan, Yang-Tian Sun, Yu-Kun Lai, Yuewen Ma, Rongfei Jia, and Lin Gao. 2022. NeRF-editing: geometry editing of neural radiance fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 18353–18364.

[66] Yu Zeng, Zhe Lin, Jimei Yang, Jianming Zhang, Eli Shechtman, and Huchuan Lu. 2020. High-resolution image inpainting with iterative confidence feedback and guided upsampling. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XIX 16*. Springer, 1–17.

[67] Haoran Zhang, Zhenzhen Hu, Changzhi Luo, Wangmeng Zuo, and Meng Wang. 2018. Semantic Image Inpainting with Progressive Generative Networks. In *ACM Multimedia*.

[68] Shengyu Zhao, Jonathan Cui, Yilun Sheng, Yue Dong, Xiao Liang, I Eric, Chao Chang, and Yan Xu. 2020. Large Scale Image Completion via Co-Modulated Generative Adversarial Networks. In *International Conference on Learning Representations*.

[69] Chuanxia Zheng, Tat-Jen Cham, and Jianfei Cai. 2019. Pluralistic image completion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 1438–1447.

[70] Chuanxia Zheng, Tat-Jen Cham, and Jianfei Cai. 2021. Pluralistic free-form image completion. *International Journal of Computer Vision* 129, 10 (2021), 2786–2805.

[71] Haitian Zheng, Zhe Lin, Jingwan Lu, Scott Cohen, Eli Shechtman, Connelly Barnes, Jianming Zhang, Ning Xu, Sohrab Amirghodsi, and Jiebo Luo. 2022. Image inpainting with cascaded modulation gan and object-aware training. In *European Conference on Computer Vision*. Springer, 277–296.