NeRF-SOS: Any-View Self-supervised Object Segmentation from Complex Real-World Scenes

Zhiwen Fan[1], Peihao Wang[1], Xinyu Gong[1], Yifan Jiang[1], Dejia Xu[1], Zhangyang Wang[1]
[1]Department of Electrical and Computer Engineering, University of Texas at Austin
{zhiwenfan,peihaowang,xinyu.gong,yifanjiang97,dejia,atlaswang}@utexas.edu

Abstract

Neural volumetric representations have shown the potential that MLP networks can be trained with multi-view calibrated images to represent scene geometry and appearance, without explicit 3D supervision. Object segmentation can enrich many downstream applications based on the learned radiance field. However, introducing hand-crafted segmentation to define regions of interest in a complex real-world scene is non-trivial and expensive as it acquires per view annotation. This paper carries out the exploration of self-supervised learning for object segmentation using NeRF for complex real-world scenes. Our framework, NeRF-SOS, couples object segmentation and neural radiance field to segment objects in any view within a scene. By proposing a novel collaborative contrastive loss in both appearance and geometry levels, NeRF-SOS encourages NeRF models to distill compact geometry-aware segmentation clusters from their density fields and the self-supervised pre-trained 2D visual features. The self-supervised object segmentation framework can be applied to various NeRF models that both lead to photo-realistic rendering results and convincing segmentations for both indoor and outdoor scenarios. Extensive results on the LLFF, Tank and Temple, and BlendedMVS datasets validate the effectiveness of NeRF-SOS. It consistently surpasses other image-based self-supervised baselines and even captures finer details than supervised Semantic-NeRF Zhi et al. (2021).

1 Introduction

Modeling the geometry of a scene is fundamental and can be applied to various real-world applications. For example, portable Augmented Reality (AR) devices (e.g., the Magic Leap One magic-leap one) reconstruct the scene geometry and localize the user by the geometry DeChicchis (2020). Despite being aware of the scene geometry, it hardly discovers the surrounding objects in the scene, and thus is difficult for the consumer to interact with the environment. The obstacles to finding and segmenting surrounding objects can be mitigated by collecting costly human-annotated data from the 3D environment, but it is not easy to deploy the system in real-world scenarios to discover objects with an ambiguous class. Therefore, enabling geometry modeling frameworks with self-supervised object segmentation would benefit many real-world applications.

Recently, neural volumetric representations for radiance fields (NeRF) Mildenhall et al. (2020a); Zhang et al. (2020); Barron et al. (2021) adopt several layers of multiple layer perceptrons (MLPs) to generate photo-realistic novel view synthesis results. NeRF and its variants utilize calibrated multi-view images to model the scene with fine details, without any explicit 3D geometry as supervision. Based on the radiance field, scene understanding has been studied by several works Vora et al. (2021); Yang et al. (2021); Zhi et al. (2021). They either require dense view annotations with heavy 3D UNet for segmentation Vora et al. (2021); Yang et al. (2021) or require human intervention to provide few semantic labels Zhi et al. (2021). Recent self-supervised discoveries of objects from the radiance field Yu et al. (2021c); Vora et al. (2021); Stelzner et al. (2021) have made attempts to decompose objects in a scene on synthetic indoor data. They introduce an additional CNN encoder, Gated Recurrent Units (GRU), and multiple NeRF models to represent all objects in both training and

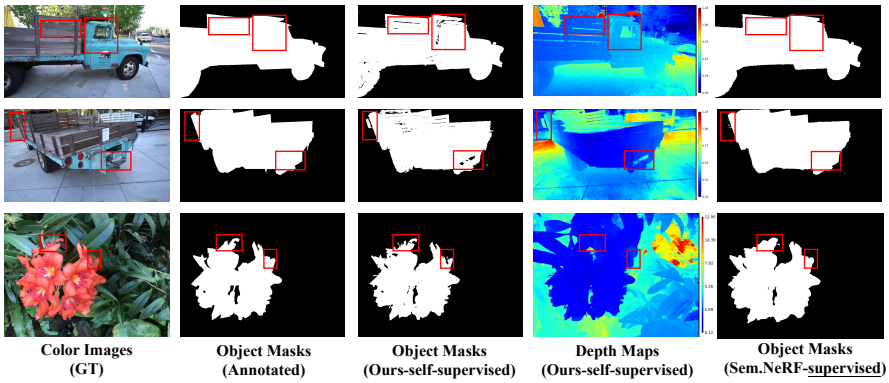

Figure 1: **Representative visual examples from the proposed self-supervised method**. We show object segmentation on new views, our method generates finer local details than supervised methods.

inference, still leaving a gap to be applied in complex real-world scenarios. In contrast to previous works, we take one step forward to investigate whether we can design a simple and general NeRF model to segment objects in real-world scenes without human intervention. Driven by this curiosity, we design a new self-supervised object segmentation framework for NeRF using a collaborative contrastive loss: by injection from the self-supervised pre-trained visual features ("**appearance level**"), and distillation from the geometry cues of a scene using NeRF's density field ("**geometry level**"). Specifically, we learn from a pre-trained 2D feature extractor in a self-supervised manner (e.g., DINO Caron et al. (2021)), and inject the visual correlations across views to form distinct feature clusters under NeRF formulation. We seek a geometry-level contrastive loss by formulating a geometric correlation volume between NeRF's density field and the segmentation clusters to make the learned feature clusters aware of scene geometry. The proposed self-supervised object segmentation framework, NeRF-SOS, is a general implicit framework and can be applied to existing NeRF models for end-to-end training. We evaluated the proposed method on the top of NeRF Mildenhall et al. (2020a) for real-world forward-facing scenes "Fortress" and "Flower", NeRF++ Zhang et al. (2020) for outdoor unbounded scene "Truck". Experiments show that NeRF-SOS significantly outperforms state-of-the-art 2D object discovery methods by producing view-consistent segmentation clusters. More surprisingly, NeRF-SOS achieves comparable segmentation accuracy with even better local details than a fully supervised trained semantic segmentation NeRF model (e.g., Semantic-NeRF Zhi et al. (2021)) shown in Figure 1, validating the proposed collaborative contrastive loss successfully injects compact feature representations and scene geometry into the object segmentor. Our contributions are summarized as follows:

- We study the self-supervised learned 2D visual feature that can be effectively distilled into the neural radiance field via an appearance contrastive loss, which forms compact feature clusters for any-view object segmentation in complex real-world scenes.
- We propose a new geometry contrastive loss for object segmentation. By leveraging its density field, our proposed framework can further inject scene geometry into the segmentation field, making the learned segmentation clusters geometry-aware.
- The proposed collaborative contrastive framework, in both appearance and geometry levels, can be implemented upon NeRF and NeRF++ for indoor and unbounded real-world scenarios. Experiments show the self-supervised object segmentation quality consistently surpasses 2D object discovery methods and yields more fine-grained segmentation results than fully-supervised Semantic-NeRF Zhi et al. (2021).

2 Related Work

Neural Radiance Fields Neural Radiance Fields (NeRF) is first proposed by Mildenhall *et al.* Mildenhall et al. (2020b), which models the underlying 3D scenes as continuous volumetric fields of color and density via layers of MLP. The input of a NeRF is a 5D vector, containing a 3D location (x, y, z) and a 2D viewing direction (θ, ϕ). Owing to NeRF's representation power and it does not require explicit geometry used to guide the training, several following works emerge trying to

address its limitations and improve the performance, such as fast training Sun et al. (2021); Deng et al. (2021), efficient inference Rebain et al. (2020); Liu et al. (2020a); Lindell et al. (2020); Garbin et al. (2021); Reiser et al. (2021); Yu et al. (2021a); Lombardi et al. (2021), better generalization Schwarz et al. (2020a); Trevithick & Yang (2020); Wang et al. (2021b); Chan et al. (2020); Yu et al. (2021b); Johari et al. (2021), supporting unconstrained scene Martin-Brualla et al. (2020); Chen et al. (2021), editing Liu et al. (2021); Jiakai et al. (2021); Wang et al. (2021a); Jang & Agapito (2021), multi-task learning Zhi et al. (2021) and view synthesis for unbounded scenes Zhang et al. (2020); Barron et al. (2021). In this paper, we treat NeRF as a powerful implicit scene representation and study how to segment objects from a complex real-world scene without any supervision.

Self-supervised 2D Image Segmentation Techniques in self-supervised feature learning Chen et al. (2020) by maximizing mutual information between an image and its augmentation can be used for self-supervised segmentation Ji et al. (2019). Successive works Li et al. (2021); Van Gansbeke et al. (2020); Cho et al. (2021); Hwang et al. (2019) either improve the clustering process or adopts Expectation-Maximization to refine segmentation. MaskContrast Van Gansbeke et al. (2021) achieves better results by contrasting learned features within and across the saliency masks, and STEGO Hamilton et al. (2022) utilizes a more powerful pre-trained model DINO-ViT Caron et al. (2021) and constructs image feature correspondence for contrastive learning. IEM Savarese et al. (2021) proposes to partition images into maximally independent sets. These methods perform well on 2D image pairs. However, it is non-trivial to implement self-supervised 2D segmentation upon neural radiance field due to the following reasons: **1).** self-supervised 2D methods failed to consider view consistency in 3D. **2).** self-supervised 2D image-based methods work on an entire image which is prohibitive for NeRF since NeRF can only render a small patch at one moment, limited by the volume rendering process Drebin et al. (1988).

Object Co-segmentation without Explicit Learning Our work aims to discover and segment visually similar object in the radiance field and, therefore, can render novel views with object masks. It is close to the object co-segmentation Rother et al. (2006) which aims to segment the common objects from a set of images Li et al. (2018). Object co-segmentation has been widely adopted in computer vision and computer graphics applications, including browsing in photo collections Rother et al. (2006), 3D reconstruction Kowdle et al. (2010), semantic segmentation Shen et al. (2017), interactive image segmentation Rother et al. (2006), object-based image retrieval Vicente et al. (2011), and video object tracking/segmentation Rother et al. (2006). Rother et al. (2006) first shows that segmenting two images outperforms the independent counterpart. This idea is analogous to the contrastive learning way in later approaches. Especially, the authors in Hénaff et al. (2022) propose the self-supervised segmentation framework using object discovery networks. Siméoni et al. (2021) localizes the objects with a self-supervised transformer. Hamilton et al. (2022) introduces the feature correspondences that distinguish between different classes. Most recently, a new co-segmentation framework based on DINO feature Amir et al. (2021) has been proposed and achieves better results on object co-segmentation and part co-segmentation.

However, extending 2D object discovery to NeRF is non-trivial as they cannot learn the geometric cues in multi-view images. uORF Yu et al. (2021c) and ObSuRF Stelzner et al. (2021) propose to use slot-based CNN encoders and object-centric latent codes for unsupervised 3D scene decomposition. Although they enable unsupervised 3D scene segmentation and novel view synthesis, experiments are on synthetic datasets, leaving a gap for complex real-world applications. Besides, a Gated Recurrent Unit (GRU) and multiple NeRF models are used, making the framework difficult to be applied to other NeRF models. In contrast, we preserve the originality of NeRF by only introducing an MLP head (a.k.a. segmentation field) on the top of NeRF models. We also design a new collaborative contrastive loss on both appearance and geometry levels. Featured by the formulation, we can distill the rich 2D visual representations and its scene geometry into compact geometry-aware segmentation clusters. The collaborative design is general and can be plug-and-play to different NeRF models.

Overview We show how to extend existing NeRF models to segment objects in both training and inference. As seen in Figure 2, we augment NeRF models by appending a parallel segmentation branch to predict point-wise implicit segmentation feature s. We propose to update the segmentation branch using a collaborative loss in both appearance and geometry levels. During inference, we preserve the segmentation branch to generate object clusters given any view by performing volume rendering Drebin et al. (1988). A clustering operation (e.g., K-means) is used to generate object masks. Specifically, NeRF-SOS inputs with multiple rays cast from several cameras, we can render the depth σ, segmentation s, and color $\boldsymbol{c}$. Next, we fed the rendered color patch $\boldsymbol{c}$ into a self-supervised pre-trained framework (e.g., DINO-ViT Caron et al. (2021)) to generate feature tensor f, constructing

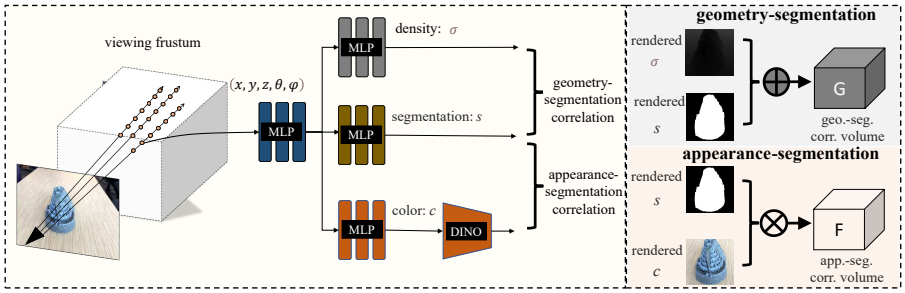

Figure 2: **The overall pipeline of the proposed NeRF-SOS.** Input with rays cast from multiple views, we render the corresponding color patch ($\boldsymbol{c}$), segmentation patch (s), and depth patch (σ). Then, appearance-segmentation correlations and geometry-segmentation correlations are used to formulate a collaborative contrastive loss, enable NeRF-SOS to render object masks from any viewpoint.

appearance-segmentation correlation volume between f and s. Similarly, a geometry-segmentation correlation volume is instantiated using σ and s. By formulating positive/negative pairs from different views, we can distill the correlation pattern in both visual feature and scene geometry into compact segmentation clusters s.

2.1 PRELIMINARIES

Neural Radiance Fields NeRF models the underlying 3D scene as a continuous volumetric radiance field of color and density. Formally, a typical radiance field can be written as $F : (\boldsymbol{x}, \boldsymbol{\theta}) \mapsto (\boldsymbol{c}, \sigma)$, where $\boldsymbol{x} \in \mathbb{R}^3$ is the spatial coordinate, $\boldsymbol{\theta} \in [-\pi, \pi]^2$ indicates the view direction, and $\boldsymbol{c} \in \mathbb{R}^3, \sigma \in \mathbb{R}_+$ represent the RGB color and density, respectively. NeRF further parameterizes this 5D-valued function by a composition of Positional Embedding (PE) and the MLP $F_\Theta = \gamma \circ \mathrm{MLP}_\Theta$, where γ is a Fourier feature mapping network Tancik et al. (2020), Θ is the network weights. Given a radiance field, NeRF follows the classical volume rendering to render an arbitrary view Max (1995).

Our goal is to fit a neural radiance from calibrated RGB images captured from multiple views. Suppose we have a set of images with corresponding extrinsic parameters. NeRF simulates the physical imaging process, by casting a ray $\boldsymbol{r} = (\boldsymbol{o}, \boldsymbol{d}, \boldsymbol{\theta})$ for each pixel via inverse perspective projection with respect to the camera pose, where $\boldsymbol{o} \in \mathbb{R}^3$ denotes the optical center of camera, $\boldsymbol{d} \in \mathbb{R}^3$ is the direction of the ray, and $\boldsymbol{\theta} \in [-\pi, \pi]^2$ is the angular view direction. We collect all pairs of rays and pixel colors as the training set $\mathcal{R} = \{(\boldsymbol{r}_i, \widehat{\boldsymbol{C}}_i)\}_{i=1}^R$, where R is the total number of rays, and $\widehat{\boldsymbol{C}}_i$ denotes the ground-truth color of the i-th ray. To simulate the color of a ray, NeRF first partitions K evenly-spaced bins between the near-far bound $[t_n, t_f]$ along the ray, and then uniformly samples one point within each bin: $t_k \sim \mathcal{U}[t_n + (k-1)(t_f - t_n)/K, t_f + k(t_f - t_n)/K]$. Afterwards, NeRF numerically evaluates volumetric ray integration Max (1995) via the following equation:

$$\boldsymbol{C}(\boldsymbol{r}|\Theta) = \sum_{k=1}^{K} T(k)(1 - \exp(-\sigma_k \Delta t_k))\boldsymbol{c}_k$$
$$\text{where } T(k) = \exp\left(-\sum_{l=1}^{k-1} \sigma_l \Delta t_l\right), \tag{1}$$

where $\Delta t_k = t_{k+1} - t_k$, and $(\boldsymbol{c}_k, \sigma_k) = F_\Theta(\boldsymbol{o} + t_k \boldsymbol{d}, \boldsymbol{\theta})$. With this forward model, NeRF optimizes the expected L_2 distance between rendered ray colors and ground-truth pixel colors as follows:

$$\mathcal{L}(\Theta|\mathcal{R}) = \mathbb{E}_{(\boldsymbol{r}, \widehat{\boldsymbol{C}}) \sim \mathbb{P}(\mathcal{R})} \left\| \boldsymbol{C}(\boldsymbol{r}|\Theta) - \widehat{\boldsymbol{C}} \right\|_2^2, \tag{2}$$

where $\mathbb{P}(\cdot)$ defines a probability measure supported in the ray space $\mathcal{R}$.

Self-supervised Learned 2D Representations DINO-ViT Caron et al. (2021) largely simplifies the self-supervised learning by applying a knowledge distillation paradigm Hinton et al. (2015) with a momentum encoder He et al. (2020), where the model is simply updated by a cross-entropy loss.

Recent works Amir et al. (2021); Hamilton et al. (2022) leverage DINO as a powerful feature extractor and proves it can learn feature correspondence for image pairs. Although DINO stands for a promising self-supervised pre-trained model with rich representations, the challenge still exists in how to leverage 2D representation to help scene understanding in the neural radiance field.

2.2 Cross View Appearance Correspondence

Visual Feature Correspondence across Views Tremendous works have explored and demonstrated the importance of object appearance when generating compact feature correspondence across views Hénaff et al. (2022); Li et al. (2018). This peculiarity is then utilized in self-supervised 2D semantic segmentation frameworks Hénaff et al. (2022); Li et al. (2018); Chen et al. (2020) to generate semantic representations, by selecting positive and negative pairs with either random or KNN-based rules Hamilton et al. (2022). Drawing inspiration from these prior arts, we construct the visual feature correspondence for NeRF at the appearance using a heuristic rule. To be more specific, we leverage the self-supervised model (e.g., DINO-ViT Caron et al. (2021)) learned from 2D image sets to distill the rich representations into compact and distinct segmentation clusters. More formally, we introduce a four-layer MLP to segment objects in the radiance field, parallel to the density branch and appearance branch of NeRF models Mildenhall et al. (2020a). Inputting the camera origin and ray cast directions, we first render multiple image patches from various viewpoints using Equation 1, then we resize them to 224 $\times$ 224 and feed them into DINO-ViT. The generated feature tensors from DINO are of $H' \times W' \times C'$) as their spatial dimensions. They are then used to generate the appearance correspondence volume Teed & Deng (2020); Hamilton et al. (2022) across views:

$$F_{hwh'w'} := \sum_{c} \frac{f_{chw}}{|f_{hw}|} \frac{f'_{ch'w'}}{|f'_{h'w'}|}, \tag{3}$$

where f and f' stand for the extracted DINO feature from two random patches in different views, (h, w) and (h', w') denote the spatial location on feature tensor for f and f', respectively. The c in the above equation denotes feature channel dimension.

Distilling Visual Feature Correspondence into Segmentation Field To inject the rich visual features f from DINO into the segmentation field $\mathcal{S}$, we formulate another segmentation correspondence volume by leveraging the predicted segmentation logits from $\mathcal{S}$ using the same rule with Equation 3. Then, we construct the appearance-segmentation correlation aims to enforce the elements of $\mathcal{S}$ and $\mathcal{S}'$ closer if f and f' are tightly coupled, where the expression with/without the superscript indicates two different views. Such correlation can be achieved via an element-wise multiplication between S and F and thereby, we have appearance contrastive loss $\mathcal{L}_{app}$:

$$\mathcal{C}_{app}(\boldsymbol{r}, b) := -\sum_{hwh'w'} (F_{hwh'w'} - b) S_{hwh'w'} \tag{4}$$

$$\mathcal{L}_{app} = \lambda_{id} \mathcal{C}_{app}(\boldsymbol{r}_{id}, b_{id}) + \lambda_{neg} \mathcal{C}_{app}(\boldsymbol{r}_{neg}, b_{neg}), \tag{5}$$

where $S_{hwh'w'} := \sum_c \frac{s_{chw}}{|s_{hw}|} \frac{s'_{ch'w'}}{|s'_{h'w'}|}$ indicates the segmentation correspondence volume between two views, $\boldsymbol{r}$ is the cast ray fed into NeRF, b is a hyper-parameter to control the positive and negative pressure. λ_{id} and λ_{neg} indicate loss force between identity pairs (positive) and distinct pairs (negative).

The intuition behind the above equation is that minimizing $\mathcal{L}$ with respect to S enforces entries in segmentation field $\mathcal{S}$ to be large when F - b are positive, and pushes entries to be small if F - b are negative.

Discover Patch Relationships via DINO To effectively find the positive/negative pairs (views) for the construction of Equation 4, we compute a similarity matrix based on the extracted [CLS] token from DINO-ViT, with the rendered image patches from c as input. The score is calculated by the cosine similarity between arbitrary patches, resulting in a $N \times N$ symmetric lookup table. An example using three patches from different views is shown in Figure **??**. The

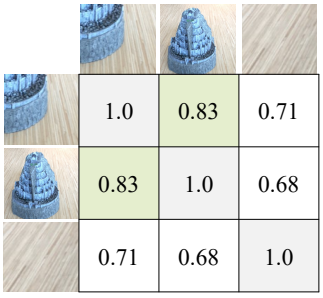

Figure 3: Rendering several patches from NeRF, we resize them to 224 $\times$ 224 and feed them to DINO. The scores are computed by mutual cosine similarities of [CLS] tokens.

intuition is that [CLS] token captures a high-level semantic appearance after self-supervised pre-training Tumanyan et al. (2022) and we found that it can effectively discover patches' similarities within our end-to-end optimization process. In our implementation, the identity pairs in Equation 4 indicate construct volume with itself (the diagonal elements), and the negative pairs are formulated by the pair of lowest similarity scores in each row.

2.3 Cross View Geometry Correspondence

Geometry Correspondence across Views With the appearance level distillation from the DINO feature to the low-dimensional segmentation embedding in the segmentation field $\mathcal{S}$, we can successfully distinguish the salient object with a similar appearance. However, $\mathcal{S}$ may mistakenly cluster different objects together, as $\mathcal{S}$ may be obfuscated by objects with distinct spatial locations but similar appearances. To solve this problem, we propose to leverage the density field that already exists in NeRF models to formulate a new geometry contrastive loss. Specifically, given a batch of K cast ray $\boldsymbol{r}$ as NeRF's input, we can obtain the density field of size $K \times N$ where N indicates the number of sampled points along each ray. By accumulating the discrete bins along each ray, we can roughly represent the density field as a single 3D point:

$$\boldsymbol{p} = \boldsymbol{r}_o + \boldsymbol{r}_d \cdot D \tag{6}$$

$$D(\boldsymbol{r}|\Theta) = \sum_{k=1}^{K} T(k)(1 - \exp(-\sigma_k \Delta t_k))t_k \tag{7}$$

where $\boldsymbol{p}$ is the accumulated 3D point along the ray, D is the estimated depth value of the corresponding pixel index. Inspired by Point Transformer Zhao et al. (2021) which uses point-wise distance as representation, we utilize the estimated point position as a geometry cue to formulate a new geometry level correspondence volume across views by measuring point-wise absolute distance:

$$G_{hwh'w'} := \sum_{c} \frac{1}{|g_{chw} - g'_{ch'w'}| + \epsilon} \tag{8}$$

where g and g' are the estimated 3D point positions in two random patches of different views, (h, w) and (h', w') denote the spatial location on feature tensor for g and g', respectively.

Injecting Geometry Coherence into Segmentation Field To inject the geometry cue from density field to segmentation field $\mathcal{S}$, we formulate segmentation correspondence volume S and geometric correspondence volume G using the same rule of Equation 4. By pulling/pushing positive/negative pairs, we come up with a new geometry-aware contrastive loss $\mathcal{L}_{geo}$ using G and S:

$$\mathcal{C}_{geo}(\boldsymbol{r}, b) := -\sum_{hwh'w'} (G_{hwh'w'} - b) S_{hwh'w'} \tag{9}$$

$$\mathcal{L}_{geo} = \lambda_{id} \mathcal{C}_{geo}(\boldsymbol{r}_{id}, b_{id}) + \lambda_{neg} \mathcal{C}_{geo}(\boldsymbol{r}_{neg}, b_{neg}) \tag{10}$$

Same as appearance contrastive loss, we find identity (positive) pairs and negative pairs via the pair-wise cosine similarity of [CLS] tokens.

2.4 Optimizing with Stride Ray Sampling

Neural Radiance Field casts a number of rays (typically not adjacent) from camera origin, intersecting the pixel, to generate input 3D points in the viewing frustum. Our model requires patch-wise rendering of size (K, K) to formulate the collaborative contrastive loss. However, we can only render a patch less than 64×64 in each view caused by GPU memory bottleneck Garbin et al. (2021). Thus, it hardly covers a sufficient receptive field to capture the global context. To solve this problem, we adopt a *Strided Ray Sampling* strategy Schwarz et al. (2020b); Meng et al. (2021), to enlarge the receptive field of the patches to capture a more global context while keeping computational cost fixed. Specifically, instead of sampling a patch of adjacent locations $M \times M$, we sample rays with an interval k, resulting in a receptive field of $(M \times k) \times (M \times k)$. Then, we optimize the overall pipeline using a balanced loss function:

$$\mathcal{L} = \lambda_0 \times \boldsymbol{C}(\boldsymbol{r}|\boldsymbol{\Theta}) + \lambda_1 \times \mathcal{L}_{app} + \lambda_2 \times \mathcal{L}_{geo}, \tag{11}$$

where λ_0, λ_1, and λ_2 are balancing weights.

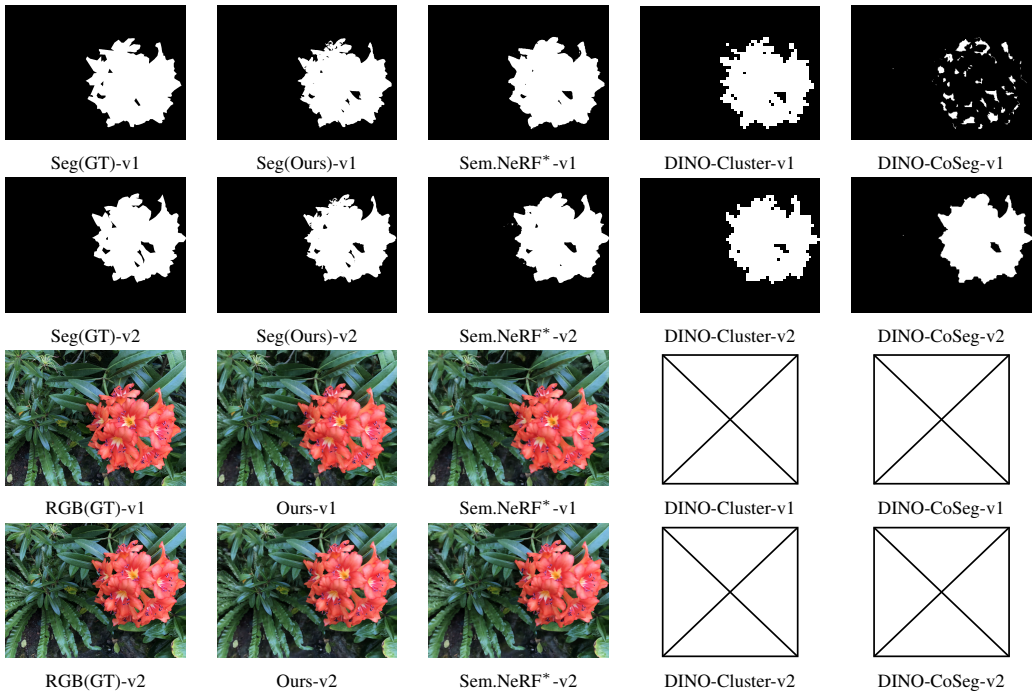

Figure 4: Qualitative results on scene Flower. As we can see in DINO-CoSeg, several discrete patches are mistakenly segmented. It is caused by DINO having higher activation on just a few tokens, which may lead to view-inconsistent and disconnected co-segmentation results. $*$ superscript denotes the supervised method using GT masks for training.

3 Experiments

3.1 Experiment setup

Datasets We run experiments and evaluate all methods on three representative datasets: Local Light Field Fusion (LLFF) dataset Mildenhall et al. (2019), Tank and Temples (T&T) dataset Riegler & Koltun (2020) abd BlendedMVS Yao et al. (2020). Particularly, we use the forward-facing scenes {"flower", "fortress"} from LLFF dataset, scene "Statue" with multiple objects from BlendedMVS dataset, and unbounded scene "Truck" from hand-held 360° captures large-scale scenes. We choose the four representative scenes because they contain at least one common object among most views. We manually labeled all test views as a binary mask to provide a fair comparison for all methods and used them to train Semantic-NeRF. Foreground objects appearing in most views are labeled as 1, while others are labeled as 0. The camera poses of scene "Truck" are estimated by COLMAP SfM Schonberger & Frahm (2016) and are processed by NeRF++ Zhang et al. (2020). We train our model on LLFF with resolution 1008 $\times$ 756 resolution, T&T with 980 $\times$ 546, and BlendedMVS with resolution 768 $\times$ 576. Train and test splits follow NeRF Mildenhall et al. (2020a) and NeRF++ Zhang et al. (2020).

3.2 Experiment results

Training Details We first implement the collaborative contrastive loss upon the original NeRF model Mildenhall et al. (2020a). In training, we first train NeRF-SOS without segmentation branch following the NeRF training recipe Mildenhall et al. (2020b) for 150k iterations. Next, we load the weight and start to train the segmentation branch alone using patch-stride ray sampling for another 50k iterations. The loss weights λ_0, λ_1, λ_2, λ_{id}, and λ_{neg} are set 0, 1, 0.01, 1 and 1 in training the segmentation branch. The segmentation branch is formulated as a four-layer MLP with ReLU as the activation function. The dimensions of hidden layers and output layers are set as 256 and 2, respectively. The segmentation results are based on K-means clustering on the segmentation logits before $\mathrm{softmax}$ layer. We train semantic-NeRF Zhi et al. (2021) 200k in total for fair comparisons. We randomly sample eight patches from different viewpoints (a.k.a batch size is 8) in training. The patch size of each sample is set as 64 $\times$ 64, with the patch stride as 6. We use the official

Table 1: Quantitative results of the novel view object segmentation results on scene "Flower". We compare NeRF-SOS with several 2D object discovery frameworks and the supervised Semantic-NeRF. To compare with the 2D object discovery methods, we render the novel views in advance to i. conduct clustering on DINO feature (DINO-Cluster), ii. DINO-CoSeg Amir et al. (2021), and iii. DOCS Li et al. (2018).

Scene "*Flower*"	PSNR ↑	SSIM ↑	LPIPS ↓	NV-ARI ↑	IoU(BG) ↑	IoU(FG) ↑	mIoU ↑
DINO+Cluster Caron et al. (2021)	-	-	-	0.8951	0.9701	0.8933	0.9317
DOCS Li et al. (2018)	-	-	-	0.0097	0.4824	0.2461	0.3643
DINO+CoSeg Amir et al. (2021)	-	-	-	0.5946	0.9036	0.5961	0.7498
NeRF-SOS (Ours)	25.96	0.7717	0.1502	0.9529	0.9869	0.9503	0.9686
Semantic-NeRF Zhi et al. (2021) (Supervised)	25.52	0.7500	0.1739	0.9104	0.9743	0.9090	0.9417

Table 2: Quantitative results of the co-segmentation results on scene "Fortress".

Scene "*Fortress*"	PSNR ↑	SSIM ↑	LPIPS ↓	NV-ARI ↑	IoU(BG) ↑	IoU(FG) ↑	mIoU ↑
DINO+Cluster Caron et al. (2021)	-	-	-	0.4939	0.8200	0.5612	0.6905
DOCS Li et al. (2018)	-	-	-	0.7412	0.9329	0.7265	0.8297
DINO+CoSeg Amir et al. (2021)	-	-	-	0.9503	0.9886	0.9395	0.9640
NeRF-SOS (Ours)	29.78	0.8517	0.1079	0.9802	0.9955	0.9751	0.9853
Semantic-NeRF Zhi et al. (2021) (Supervised)	29.78	0.8578	0.0906	0.9838	0.9963	0.9799	0.9881

pre-trained DINO-ViT in a self-supervised manner on ImageNet dataset as our 2D feature extractor. The pre-trained DINO backbone is kept frozen for all layers during training. All hyperparameters are carefully tuned by a grid search, and the best configuration is applied to all experiments. All models are trained on an NVIDIA RTX A6000 GPU with 48 GB memory.

Implementation of the Patch Selection We reconstruct positive pairs and negative pairs on the fly during training. Given N rendered patches from N different viewpoints in training, we fed the patches into the DINO-ViT and obtained the [CLS] tokens. Next, we compute a $N \times N$ similarity matrix using the cosine similarity with the N [CLS] tokens. The negative pairs are selected from the pair with the lowest similarity in each row; the positive pairs are set as the identity pairs. Overall, $2\times N$ pairs (N positives + N negatives) will be formulated in each iteration to compute the collaborative contrastive loss.

Metrics We adopt the Adjusted Rand Index (ARI) as our metric to evaluate the clustering quality. As we only consider the ARI in novel views, we report it as NV-ARI. We also adopt mean Intersection-over-Union to measure segmentation quality for both object and background, as we set the clusters with larger activation as foreground by DINO. To evaluate the rendering quality, we follow NeRF Mildenhall et al. (2020a), adopting signal-to-noise ratio (PSNR), the structural similarity index measure (SSIM) Wang et al. (2004) and learned perceptual image patch similarity (LPIPS) Zhang et al. (2018) as evaluation metrics.

Self-supervised Object Segmentation on LLFF We build NeRF-SOS on the vanilla NeRF Mildenhall et al. (2020a) to validate its effectiveness on LLFF datasets. Two groups of current object segmentation are adopted for comparisons: i. NeRF-based methods, including our NeRF-SOS, and supervised Semantic-NeRF Zhi et al. (2021) trained with GT masks; and ii. image-based object co-segmentation methods: DINO-CoSeg Amir et al. (2021) and DINO-ViT with K-means clustering Caron et al. (2021) and DOCS Li et al. (2018). As image-based co-segmentation methods cannot evaluate novel views, we pre-render the new views using NeRF and construct image pairs between the first image in the test set with others for DINO-CoSeg Amir et al. (2021) and DOCS Li et al. (2018). The evaluations on DINO-ViT Caron et al. (2021) clustering also use the pre-rendered images, obtaining its DINO feature in the last layer and performing K-means to identify clusters of data objects.

Quantitative comparisons against segmentation methods on scene *Flower* are provided in Table 1, together with qualitative visualizations shown in Figure 4. Here, we visualize two different views to show segmentation consistency across views. The quantitative evaluations metrics on scene *Fortress* are shown in Table 2. These results convey several observations to us: **1).** NeRF-SOS consistently outperforms image-based co-segmentation in evaluation metrics and view-consistency. **2).** Compared with SoTA supervised NeRF segmentation method (Semantic-NeRF Zhi et al. (2021)), our method effectively segments the object within the scene and performs on par in both evaluation metrics and visualization.

Self-supervised Object Segmentation on Unbounded Scene To test the generalization ability of the proposed collaborative contrastive loss, we implement it on NeRF++ Zhang et al. (2020) to test with a more challenging unbounded scene. Here, we mainly evaluate all previously mentioned

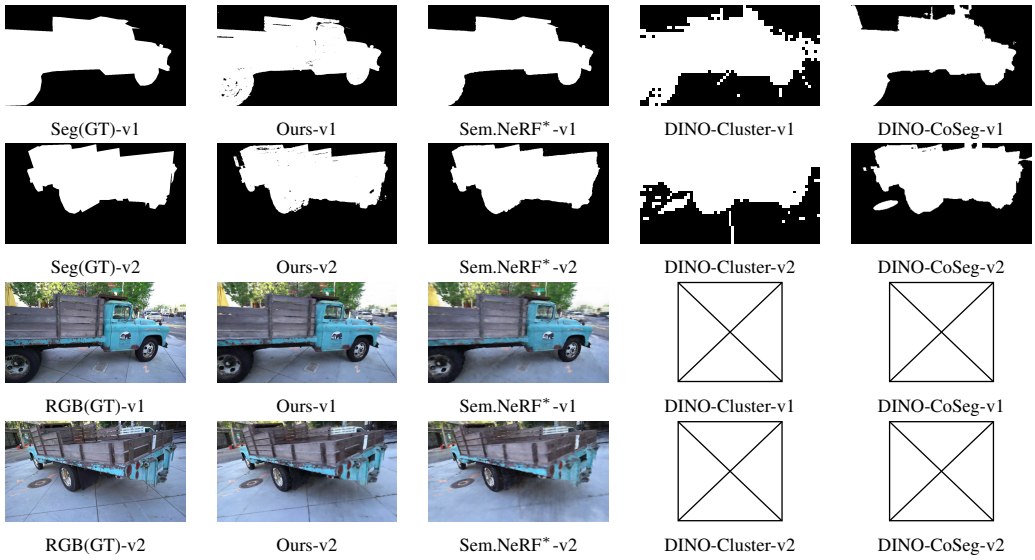

Figure 5: Qualitative novel view object segmentation results on unbounded scene *Truck*. NeRF-SOS (the second column) produces view-consistent masks more than other self-supervised methods. It even generates finer details than supervised Semantic-NeRF++ when the ground truth masks are not perfect (see the gaps between wooden slats in view1, and the side view mirror in view2). $*$ superscript denotes the supervised method using GT masks for training.

Table 3: Quantitative results of the object segmentation results on outdoor unbounded scene "Truck". We compare it with supervised Semantic-NeRF to show that our self-supervised NeRF-SOS performs on par with the supervised methods. To compare with the image-based co-segmentation method, we adopt DINO+Clustering, DINO-CoSeg Amir et al. (2021) and DOCS Li et al. (2018) by rendering the novel views in advance.

Scene "*Truck*"	PSNR ↑	SSIM ↑	LPIPS ↓	NV-ARI ↑	IoU(BG) ↑	IoU(FG) ↑	mIoU ↑
DINO+Cluster Caron et al. (2021)	-	-	-	0.2937	0.6239	0.6153	0.6196
DOCS Li et al. (2018)	-	-	-	0.1517	0.6845	0.2463	0.4654
DINO+CoSeg Amir et al. (2021)	-	-	-	0.8571	0.9408	0.9080	0.9244
NeRF-SOS (Ours)	22.20	0.7000	0.2691	0.9207	0.9689	0.9455	0.9572
Semantic-NeRF++ Zhi et al. (2021) (Supervised)	21.08	0.6350	0.4114	0.9674	0.9869	0.9782	0.9826

methods on scene "Truck" as it is the only scene captured surrounding an object provided by NeRF++. We re-implement Semantic-NeRF using NeRF++ as the backbone model for unbounded setting, termed Semantic-NeRF++, following the training recipe of NeRF++. Qualitative and quantitative results are shown in Figure 5 and Table 3. We can see that our NeRF-SOS still surpasses state-of-the-art image-based object co-segmentation methods. Compared with supervised Semantic-NeRF++, NeRF-SOS achieves slightly worse results on evaluation metrics. However, if we dive into the visualizations, we see that NeRF-SOS generates surprising segmentation quality. For example, **1).** In Figure 5(Ours-v2), NeRF-SOS can recognize the side view mirror adjacent to the truck even when the region is mistakenly annotated in the ground truth masks. **2).** In Figure 5(Ours-v1) NeRF-SOS can distinguish the gaps between the wooden slats as the gaps have distinct depth values than the neighboring slats, thanks to the contribution of geometry-aware contrastive loss.

Impact of the Geometry Contrastive Loss To understand the impact of the geometry contrastive loss, we perform experiments on unbounded scene *Truck* and report results in Figure 9. In this part, we set two baseline models, either using appearance or geometric contrastive loss alone on NeRF++, as visual features and its geometry can all serve as cues for object segmentation. We can see that without geometric constraints (5th row), the segmentation branch failed to cluster spatially continuous objects; without visual cues (6th row), the model lost the perception of the truck. In contrast, a collaborative loss generates convincing object segmentation.

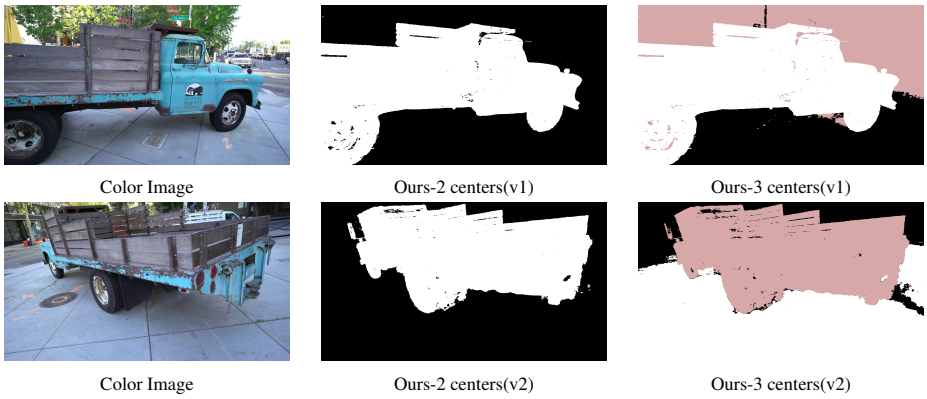

Figure 6: Qualitative results on scene Truck with different cluster centers on its distilled segmentation field. Note that, the cross-view visualized colors of multiple-center clustering are not corresponding to the subject ID, as we perform unsupervised clustering.

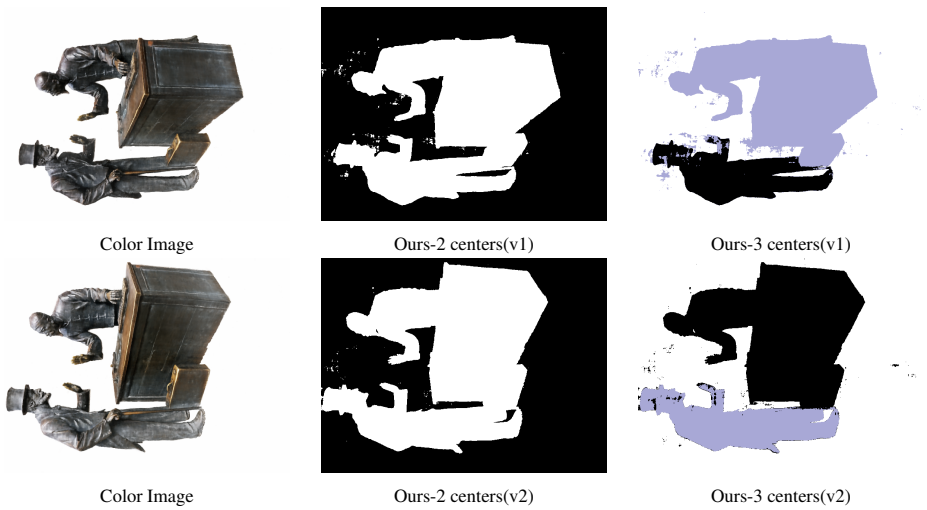

Figure 7: Qualitative results on scene Statues with different cluster center on its distilled segmentation field. Note that, the cross-view visualized colors of multiple-center clustering are not corresponding to the subject ID, as we perform unsupervised clustering. The floaters are mainly caused by the learned incorrect scene geometry.

3.3 Qualitative Results of More Cluster Centers

We perform the study by performing multiple-center clusters on the distilled segmentation field in Figure 6. As can be seen, regions with distinct appearances and physical distances are separated into different clusters. Qualitative visualization on a subset of BlendedMVS Yao et al. (2020) (scene *Statue*, pre-processing by NSVF Liu et al. (2020b)) are provided in Figure 7. Here, we visualize two different views to show segmentation results, with cluster=2 and cluster=3. The noisy floaters are mainly caused by the incorrect scene geometry learned by NeRF, which we will discuss later.

Evaluation Details The hyperparameters of NeRF-SOS at different scenes are shown in Table 4.

4 Failed Cases

As we perform clustering on the learned segmentation field, it potentially generates inconsistent masks cross views. The floaters shown in Figure 8 that appear in the object masks are highly correlated with the incorrect background geometry learned by NeRF, as the modeling of real-world geometry is extremely challenging.

Table 4: hyperparameters of NeRF-SOS on different scenes.

parameter name	Scene *Flower*	Scene *Fortress*	Scene *Truck*
$\lambda_{id}(\mathcal{L}_{app})$	1.00	1.00	1.00
$\lambda_{neg}(\mathcal{L}_{app})$	1.00	1.00	1.00
$b_{id}(\mathcal{L}_{app})$	0.18	0.18	0.18
$b_{neg}(\mathcal{L}_{app})$	0.46	0.46	0.46
$\lambda_{id}(\mathcal{L}_{geo})$	1.00	1.00	1.00
$\lambda_{neg}(\mathcal{L}_{geo})$	1.00	1.00	1.00
$b_{id}(\mathcal{L}_{geo})$	0.50	0.50	1.00
$b_{neg}(\mathcal{L}_{geo})$	3.00	3.00	5.00

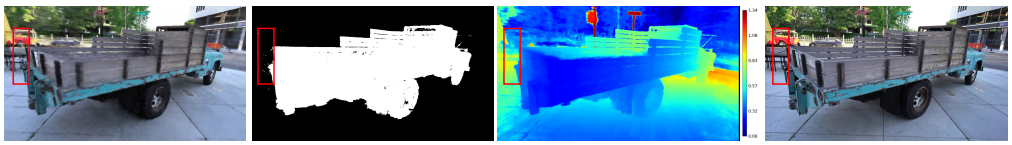

Figure 8: Failed case on object boundary caused by imperfect learned geometry of NeRF model.

5 Conclusion, Discussion of Limitation and Broader Impact

We present NeRF-SOS, a framework that learns object segmentation for any view from complex real-world scenes. NeRF-SOS is based on a self-supervised framework, where a collaborative contrastive loss in appearance-segmentation and geometry-segmentation levels are included. Comprehensive experiments on forward-facing scenes and unbounded scenes are conducted with SoTA image-based object segmentation frameworks and fully supervised Semantic-NeRF. We found that NeRF-SOS consistently performs better than image-based methods and sometimes generates finer segmentation details than supervised segmentors. However, similar to other scene-specific NeRF methods, one limitation of NeRF-SOS is that it cannot segment across scenes, which we will explore in our future works. For broader impact, this work first attempts a self-supervised 3D object segmentation framework for complex real-world scenes using NeRF, which reduces the annotation and energy cost required by ML models.

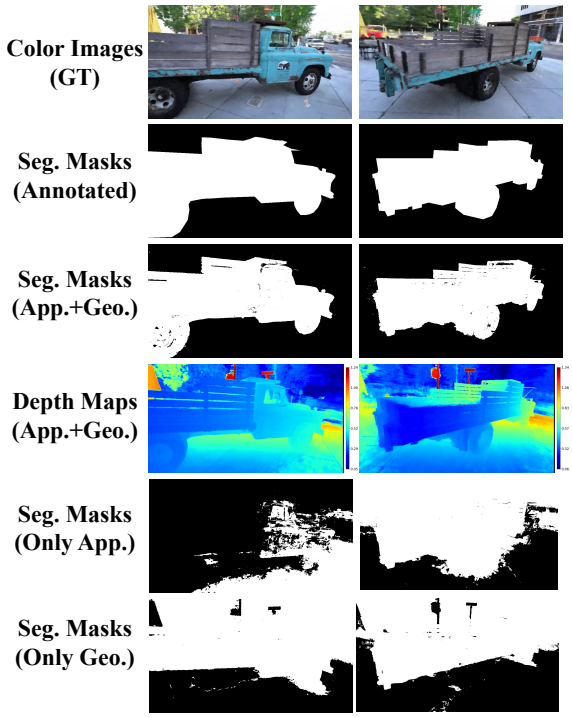

Figure 9: Segmentation of three variants using different contrastive losses is shown in rows 3, 5, and 6. APP.+Geo. indicates our collaborative loss; App. means appearance contrastive loss; Geo. means geometric contrastive.

REFERENCES

Shir Amir, Yossi Gandelsman, Shai Bagon, and Tali Dekel. Deep vit features as dense visual descriptors. *arXiv preprint arXiv:2112.05814*, 2021.

Jonathan T Barron, Ben Mildenhall, Matthew Tancik, Peter Hedman, Ricardo Martin-Brualla, and Pratul P Srinivasan. Mip-nerf: A multiscale representation for anti-aliasing neural radiance fields. In *IEEE International Conference on Computer Vision (ICCV)*, 2021.

Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 9650–9660, 2021.

Eric Chan, Marco Monteiro, Peter Kellnhofer, Jiajun Wu, and Gordon Wetzstein. pi-GAN: Periodic implicit generative adversarial networks for 3D-aware image synthesis. *https://arxiv.org/abs/2012.00926*, 2020.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pp. 1597–1607. PMLR, 2020.

Xingyu Chen, Qi Zhang, Xiaoyu Li, Yue Chen, Feng Ying, Xuan Wang, and Jue Wang. Hallucinated neural radiance fields in the wild, 2021.

Jang Hyun Cho, Utkarsh Mall, Kavita Bala, and Bharath Hariharan. Picie: Unsupervised semantic segmentation using invariance and equivariance in clustering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 16794–16804, 2021.

Joseph DeChicchis. Semantic understanding for augmented reality and its applications. 2020.

Kangle Deng, Andrew Liu, Jun-Yan Zhu, and Deva Ramanan. Depth-supervised nerf: Fewer views and faster training for free. *arXiv preprint arXiv:2107.02791*, 2021.

Robert A Drebin, Loren Carpenter, and Pat Hanrahan. Volume rendering. *ACM Siggraph Computer Graphics*, 22(4):65–74, 1988.

Stephan J. Garbin, Marek Kowalski, Matthew Johnson, Jamie Shotton, and Julien Valentin. Fastnerf: High-fidelity neural rendering at 200fps. *https://arxiv.org/abs/2103.10380*, 2021.

Mark Hamilton, Zhoutong Zhang, Bharath Hariharan, Noah Snavely, and William T Freeman. Unsupervised semantic segmentation by distilling feature correspondences. *arXiv preprint arXiv:2203.08414*, 2022.

Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 9729–9738, 2020.

Olivier J Hénaff, Skanda Koppula, Evan Shelhamer, Daniel Zoran, Andrew Jaegle, Andrew Zisserman, João Carreira, and Relja Arandjelović. Object discovery and representation networks. *arXiv preprint arXiv:2203.08777*, 2022.

Geoffrey Hinton, Oriol Vinyals, Jeff Dean, et al. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2(7), 2015.

Jyh-Jing Hwang, Stella X Yu, Jianbo Shi, Maxwell D Collins, Tien-Ju Yang, Xiao Zhang, and Liang-Chieh Chen. Segsort: Segmentation by discriminative sorting of segments. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 7334–7344, 2019.

Wonbong Jang and Lourdes Agapito. Codenerf: Disentangled neural radiance fields for object categories. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 12949–12958, 2021.

Xu Ji, Joao F Henriques, and Andrea Vedaldi. Invariant information clustering for unsupervised image classification and segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 9865–9874, 2019.

Zhang Jiakai, Liu Xinhang, Ye Xinyi, Zhao Fuqiang, Zhang Yanshun, Wu Minye, Zhang Yingliang, Xu Lan, and Yu Jingyi. Editable free-viewpoint video using a layered neural representation. In *ACM SIGGRAPH*, 2021.

Mohammad Mahdi Johari, Yann Lepoittevin, and François Fleuret. Geonerf: Generalizing nerf with geometry priors. *arXiv preprint arXiv:2111.13539*, 2021.

Adarsh Kowdle, Dhruv Batra, Wen-Chao Chen, and Tsuhan Chen. imodel: interactive co-segmentation for object of interest 3d modeling. In *European Conference on Computer Vision*, pp. 211–224. Springer, 2010.

Weihao Li, Omid Hosseini Jafari, and Carsten Rother. Deep object co-segmentation. In *Asian Conference on Computer Vision*, pp. 638–653. Springer, 2018.

Yunfan Li, Peng Hu, Zitao Liu, Dezhong Peng, Joey Tianyi Zhou, and Xi Peng. Contrastive clustering. In *2021 AAAI Conference on Artificial Intelligence (AAAI)*, 2021.

David Lindell, Julien Martel, and Gordon Wetzstein. AutoInt: Automatic integration for fast neural volume rendering. *https://arxiv.org/abs/2012.01714*, 2020.

Lingjie Liu, Jiatao Gu, Kyaw Zaw Lin, Tat-Seng Chua, and Christian Theobalt. Neural sparse voxel fields. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 33, 2020a.

Lingjie Liu, Jiatao Gu, Kyaw Zaw Lin, Tat-Seng Chua, and Christian Theobalt. Neural sparse voxel fields. *Advances in Neural Information Processing Systems*, 33:15651–15663, 2020b.

Steven Liu, Xiuming Zhang, Zhoutong Zhang, Richard Zhang, Jun-Yan Zhu, and Bryan Russell. Editing conditional radiance fields, 2021.

Stephen Lombardi, Tomas Simon, Gabriel Schwartz, Michael Zollhoefer, Yaser Sheikh, and Jason Saragih. Mixture of volumetric primitives for efficient neural rendering, 2021.

magic-leap one. magic-leap-one. `https://www.magicleap.com/magic-leap-one`.

Ricardo Martin-Brualla, Noha Radwan, Mehdi Sajjadi, Jonathan T. Barron, Alexey Dosovitskiy, and Daniel Duckworth. NeRF in the wild: Neural radiance fields for unconstrained photo collections. *https://arxiv.org/abs/2008.02268*, 2020.

Nelson Max. Optical models for direct volume rendering. *IEEE Transactions on Visualization and Computer Graphics (TVCG)*, 1995.

Quan Meng, Anpei Chen, Haimin Luo, Minye Wu, Hao Su, Lan Xu, Xuming He, and Jingyi Yu. Gnerf: Gan-based neural radiance field without posed camera. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 6351–6361, 2021.

Ben Mildenhall, Pratul P Srinivasan, Rodrigo Ortiz-Cayon, Nima Khademi Kalantari, Ravi Ramamoorthi, Ren Ng, and Abhishek Kar. Local light field fusion: Practical view synthesis with prescriptive sampling guidelines. *ACM Transactions on Graphics (TOG)*, 2019.

Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. In *European conference on computer vision*, pp. 405–421. Springer, 2020a.

Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. In *European conference on computer vision*, pp. 405–421. Springer, 2020b.

Daniel Rebain, Wei Jiang, Soroosh Yazdani, Ke Li, Kwang Moo Yi, and Andrea Tagliasacchi. DeRF: Decomposed radiance fields. *https://arxiv.org/abs/2011.12490*, 2020.

Christian Reiser, Songyou Peng, Yiyi Liao, and Andreas Geiger. Kilonerf: Speeding up neural radiance fields with thousands of tiny mlps. In *IEEE International Conference on Computer Vision (ICCV)*, 2021.

Gernot Riegler and Vladlen Koltun. Free view synthesis. In *European Conference on Computer Vision (ECCV)*, 2020.

Carsten Rother, Tom Minka, Andrew Blake, and Vladimir Kolmogorov. Cosegmentation of image pairs by histogram matching-incorporating a global constraint into mrfs. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, volume 1, pp. 993–1000. IEEE, 2006.

Pedro Savarese, Sunnie SY Kim, Michael Maire, Greg Shakhnarovich, and David McAllester. Information-theoretic segmentation by inpainting error maximization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4029–4039, 2021.

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

Katja Schwarz, Yiyi Liao, Michael Niemeyer, and Andreas Geiger. Graf: Generative radiance fields for 3D-aware image synthesis. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 33, 2020a.

Katja Schwarz, Yiyi Liao, Michael Niemeyer, and Andreas Geiger. Graf: Generative radiance fields for 3d-aware image synthesis. *Advances in Neural Information Processing Systems*, 33: 20154–20166, 2020b.

Tong Shen, Guosheng Lin, Lingqiao Liu, Chunhua Shen, and Ian Reid. Weakly supervised semantic segmentation based on co-segmentation. In *BMVC*, 2017.

Oriane Siméoni, Gilles Puy, Huy V Vo, Simon Roburin, Spyros Gidaris, Andrei Bursuc, Patrick Pérez, Renaud Marlet, and Jean Ponce. Localizing objects with self-supervised transformers and no labels. *arXiv preprint arXiv:2109.14279*, 2021.

Karl Stelzner, Kristian Kersting, and Adam R Kosiorek. Decomposing 3d scenes into objects via unsupervised volume segmentation. *arXiv preprint arXiv:2104.01148*, 2021.

Cheng Sun, Min Sun, and Hwann-Tzong Chen. Direct voxel grid optimization: Super-fast convergence for radiance fields reconstruction. *arXiv preprint arXiv:2111.11215*, 2021.

Matthew Tancik, Pratul P Srinivasan, Ben Mildenhall, Sara Fridovich-Keil, Nithin Raghavan, Utkarsh Singhal, Ravi Ramamoorthi, Jonathan T Barron, and Ren Ng. Fourier features let networks learn high frequency functions in low dimensional domains. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

Zachary Teed and Jia Deng. Raft: Recurrent all-pairs field transforms for optical flow. In *European conference on computer vision*, pp. 402–419. Springer, 2020.

Alex Trevithick and Bo Yang. GRF: Learning a general radiance field for 3D scene representation and rendering. *https://arxiv.org/abs/2010.04595*, 2020.

Narek Tumanyan, Omer Bar-Tal, Shai Bagon, and Tali Dekel. Splicing vit features for semantic appearance transfer. *arXiv preprint arXiv:2201.00424*, 2022.

Wouter Van Gansbeke, Simon Vandenhende, Stamatios Georgoulis, Marc Proesmans, and Luc Van Gool. Scan: Learning to classify images without labels. In *European Conference on Computer Vision*, pp. 268–285. Springer, 2020.

Wouter Van Gansbeke, Simon Vandenhende, Stamatios Georgoulis, and Luc Van Gool. Unsupervised semantic segmentation by contrasting object mask proposals. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 10052–10062, 2021.

Sara Vicente, Carsten Rother, and Vladimir Kolmogorov. Object cosegmentation. In *CVPR 2011*, pp. 2217–2224. IEEE, 2011.

Suhani Vora, Noha Radwan, Klaus Greff, Henning Meyer, Kyle Genova, Mehdi SM Sajjadi, Etienne Pot, Andrea Tagliasacchi, and Daniel Duckworth. Nesf: Neural semantic fields for generalizable semantic segmentation of 3d scenes. *arXiv preprint arXiv:2111.13260*, 2021.

Can Wang, Menglei Chai, Mingming He, Dongdong Chen, and Jing Liao. Clip-nerf: Text-and-image driven manipulation of neural radiance fields. *arXiv preprint arXiv:2112.05139*, 2021a.

Qianqian Wang, Zhicheng Wang, Kyle Genova, Pratul P Srinivasan, Howard Zhou, Jonathan T Barron, Ricardo Martin-Brualla, Noah Snavely, and Thomas Funkhouser. Ibrnet: Learning multi-view image-based rendering. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021b.

Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004.

Bangbang Yang, Yinda Zhang, Yinghao Xu, Yijin Li, Han Zhou, Hujun Bao, Guofeng Zhang, and Zhaopeng Cui. Learning object-compositional neural radiance field for editable scene rendering. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 13779–13788, 2021.

Yao Yao, Zixin Luo, Shiwei Li, Jingyang Zhang, Yufan Ren, Lei Zhou, Tian Fang, and Long Quan. Blendedmvs: A large-scale dataset for generalized multi-view stereo networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1790–1799, 2020.

Alex Yu, Ruilong Li, Matthew Tancik, Hao Li, Ren Ng, and Angjoo Kanazawa. Plenoctrees for real-time rendering of neural radiance fields. In *IEEE International Conference on Computer Vision (ICCV)*, 2021a.

Alex Yu, Vickie Ye, Matthew Tancik, and Angjoo Kanazawa. pixelnerf: Neural radiance fields from one or few images. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021b.

Hong-Xing Yu, Leonidas J Guibas, and Jiajun Wu. Unsupervised discovery of object radiance fields. *arXiv preprint arXiv:2107.07905*, 2021c.

Kai Zhang, Gernot Riegler, Noah Snavely, and Vladlen Koltun. Nerf++: Analyzing and improving neural radiance fields. *arXiv preprint arXiv:2010.07492*, 2020.

Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 586–595, 2018.

Hengshuang Zhao, Li Jiang, Jiaya Jia, Philip HS Torr, and Vladlen Koltun. Point transformer. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 16259–16268, 2021.

Shuaifeng Zhi, Tristan Laidlow, Stefan Leutenegger, and Andrew J Davison. In-place scene labelling and understanding with implicit scene representation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 15838–15847, 2021.