GP-NeRF: Generalized Perception NeRF for Context-Aware 3D Scene Understanding

Hao Li[1], Dingwen Zhang[1,6,*], Yalun Dai[4], Nian Liu[2,*], Lechao Cheng[3] , Jingfeng Li[1],
Jingdong Wang[5], Junwei Han[1,6]

[1] Brain and Artificial Intelligence Lab, Northwestern Polytechnical University [2] MBZUAI
[3] Hefei University of Technology [4] Nanyang Technological University [5] Baidu, Inc.
[6] Institute of Artificial Intelligence, Hefei Comprehensive National Science Center
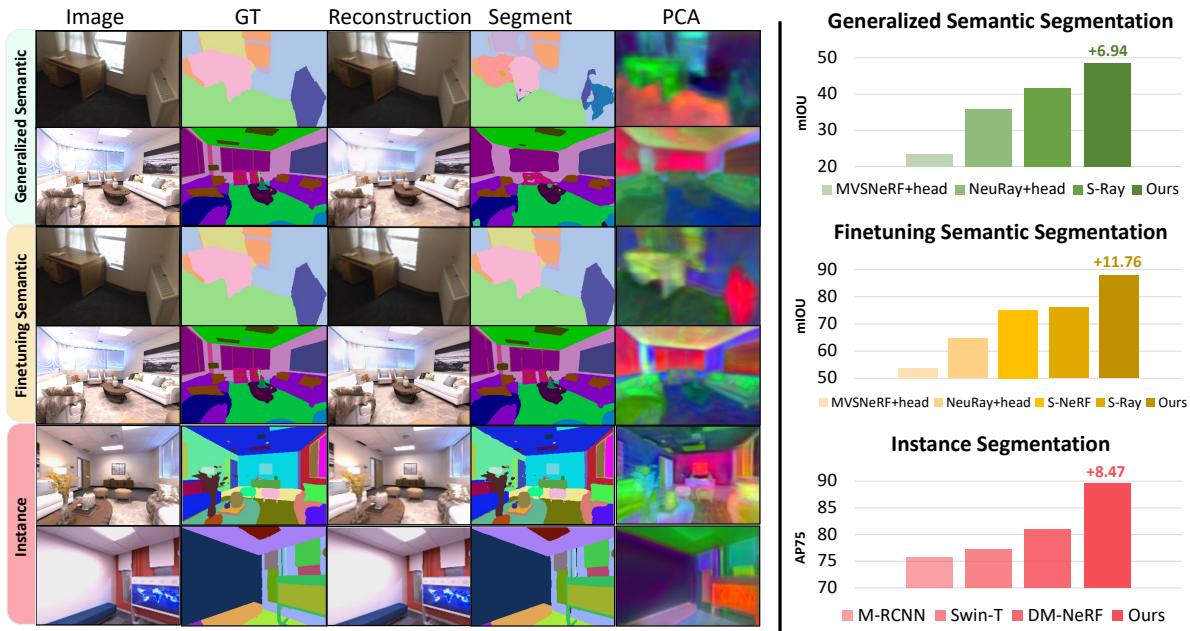* Corresponding authors

Figure 1. Our method, called **GP-NeRF**, achieves remarkable performance improvements for instance and semantic segmentation in both synthesis [35] and real-world [10] datasets, as shown in the right column of the figure. Here we showcase generalized semantic segmentation, finetuning semantic segmentation, and instance segmentation) with their corresponding reconstruction results. For the left column, the qualitative results of the visualization are presented, showing the effectiveness of our method for simultaneous segmentation and reconstruction. What's more, we visualize our rendered features via PCA in the novel view, demonstrating our method possesses the capability to produce semantic-aware features that can distinguish between different classes and objects.

Abstract

Applying Neural Radiance Fields (NeRF) to downstream perception tasks for scene understanding and representation is becoming increasingly popular. Most existing methods treat semantic prediction as an additional rendering task, i.e., the "label rendering" task, to build semantic NeRFs. However, by rendering semantic/instance labels per pixel without considering the contextual information of the rendered image, these methods usually suffer from unclear boundary segmentation and abnormal segmentation of pix-

els within an object. To solve this problem, we propose Generalized Perception NeRF (GP-NeRF), a novel pipeline that makes the widely used segmentation model and NeRF work compatibly under a unified framework, for facilitating context-aware 3D scene perception. To accomplish this goal, we introduce transformers to aggregate radiance as well as semantic embedding fields jointly for novel views and facilitate the joint volumetric rendering of both fields. In addition, we propose two self-distillation mechanisms, i.e., the Semantic Distill Loss and the Depth-Guided Semantic Distill Loss, to enhance the discrimination and qual-

ity of the semantic field and the maintenance of geometric consistency. In evaluation, as shown in Fig. 1 we conduct experimental comparisons under two perception tasks (i.e. semantic and instance segmentation) using both synthetic and real-world datasets. Notably, our method outperforms SOTA approaches by 6.94%, 11.76%, and 8.47% on generalized semantic segmentation, finetuning semantic segmentation, and instance segmentation, respectively. Project.

1. Introduction

Robust scene understanding models are crucial for enabling various applications, including virtual reality (VR) [17], robot navigation [48], self-driving [12], and more [1]. They have experienced tremendous progress over the past years, driven by continuously improved model architectures [7, 8, 24, 26, 53] in 2D image segmentation. However, these methods face challenges due to their lack of specific scene representation and the inability to track unique object identities across different views [34].

Meanwhile, implicit neural representations [27, 28, 38, 41] have demonstrated an impressive capability in capturing the 3D structure of complex real-world scenes [10]. By adopting multi-layer perceptions, it utilizes multi-view images to learn 3D representations for synthesizing images in novel views with fine-grained details. This success has spurred research into applying NeRF for robust scene understanding, aiming to explore a broader range of possibilities in high-level vision tasks and applications.

Recent works [13, 22, 34, 54] addressed scene understanding from 2D images by exploring semantics using Neural Radiance Fields (NeRFs) [28]. Per-scene optimized methods, such as Semantic-NeRF [54], DM-NeRF [40], and Panoptic-NeRF [13], simply utilize additional Multi-Layer Perceptron (MLP) to regress the semantic class for each 3D-point together with radiance and density. The latest method Semantic-Ray [22], based on generalized NeRF NeuRay [27], achieves generalized semantic segmentation by introducing an individual learnable semantic branch to construct the semantic field and render semantic features in novel view using frozen density.

Although this operation is reasonable to build a semantic field, it falls short in achieving joint optimization of both RGB rendering and semantic prediction, thus missing an important message when building high-quality heterogeneous embedding fields: The geometry distribution of the radiance field and Semantic-Embedding field should be consistent with each other. For example: 1) The boundaries of different objects are usually distinct in RGB representation, they could be utilized for achieving more accurate boundary segmentation; and 2) The areas belonging to the same object often share consistent coloration, which can act as informative cues to enhance the quality of RGB recon-

struction. Moreover, Semantic-Ray follows the vanilla semantic NeRF by rendering semantic labels for each point independently in the novel view, ignoring the context information, such as the relationships and interactions between the nearby pixels and objects.

To address these problems, we present Generalized Perception NeRF (GP-NeRF), a novel unified learning framework that embeds NeRF and the powerful 2D segmentation modules together to perform context-aware 3D scene perception. As shown in Fig. 2, GP-NeRF utilizes Field Aggregation Transformer to aggregate the radiance field as well as the semantic-embedding field, and Ray Aggregation Transformer to render them jointly in novel views. Both processes perform under a joint optimization scheme. Specifically, we render rich-semantic features rather than labels in novel views and feed them into a powerful 2D segmentation module to perform context-aware semantic perception. To enable our framework to work compatibly, we further introduce two novel self-distillation mechanisms: 1) the Semantic Distill Loss, which enhances the discrimination and quality of the semantic field, thereby facilitating improved prediction performance by the perception head; and 2) the Depth-Guided Semantic Distill Loss, which aims to supervise the semantic representation of each point within the semantic field, ensuring the maintenance of geometric consistency. Under such mechanisms, our method bridges the gap between the powerful 2D segmentation modules and NeRF methods, offering a possible integration solution with existing downstream perception heads.

Our contributions can be summarized as follows:

- We make an early effort to establish a unified learning framework that can combine NeRF and segmentation modules to perform context-aware 3D scene perception.
- Technically, we use Transformers to jointly construct radiance as well as semantic embedding fields and facilitate the joint volumetric rendering upon both fields for novel views.
- The 2D and depth-guided self-distillation mechanisms are proposed to boost the discrimination and quality of the semantic embedding field.
- Comprehensive experiments are conducted. The results demonstrate that our method can surpass existing NeRF methods in downstream perception tasks (*i.e.* semantic, instance) with both generalized and per-scene settings.

2. Related Work

2.1. Neural Radiance Fields (NeRF)

Neural Radiance Fields (NeRF), introduced by Mildenhall et al. [28], have revolutionized view synthesis by fitting scenes into a continuous 5D radiance field using MLPs. Subsequent enhancements include Mip-NeRF's [2, 3] efficient scaling in unbounded scenes, Nex's [43] handling of
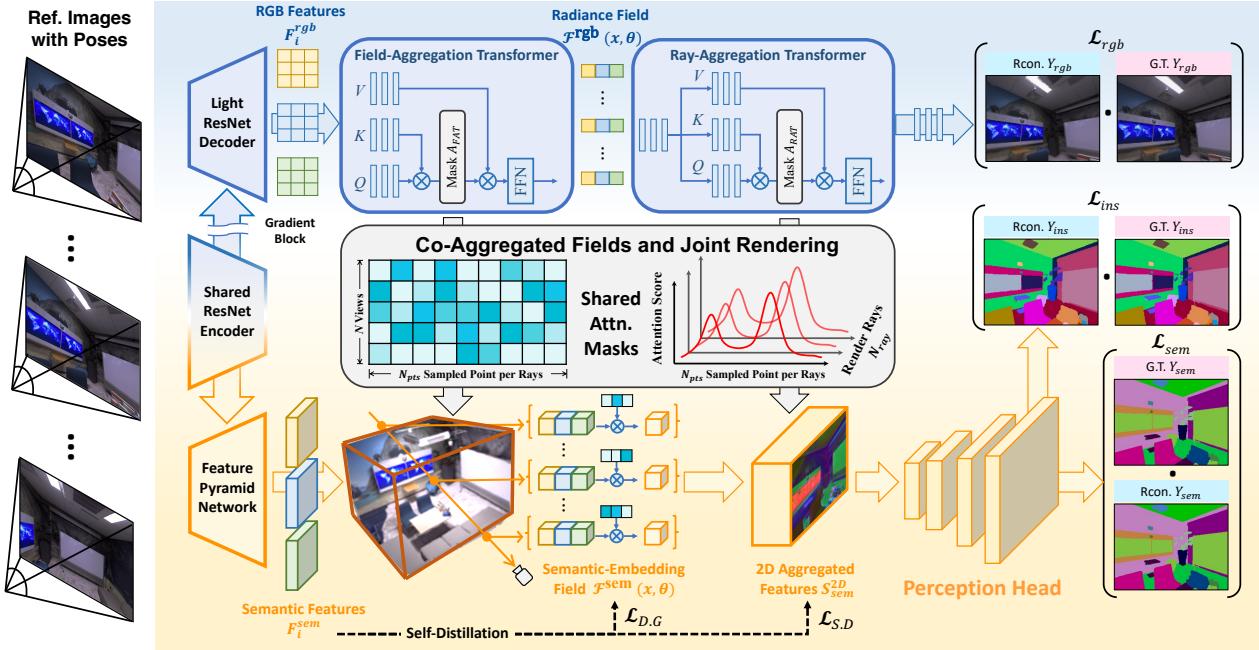
Figure 2. **Overview of proposed GP-NeRF**. Given reference views with their poses, we embed NeRF into the segmenter to perform context-aware semantic Y_{sem} /instance Y_{ins} segmentation and ray reconstruction Y_{rgb} in novel view (Sec. 4.1). In detail, we use Transformers to co-aggregate Radiance as well as Semantic-Embedding fields and render them jointly in novel views (Sec. 4.2). Specifically, we propose two self-distillation mechanisms to boost the discrimination and quality of the semantic embedding field (Sec. 4.3).

large view-dependent effects, improvements in surface representation [29, 46] and dynamic scene adaptation [30, 31], as well as advancements in lighting, reflection [6, 39], and depth-based regression [11, 44]. Methods like PixelNeRF [49], IBRNet [41], NeuRay [27], and GNT [38] further reduce the need for per-scene training by using cross-scene multi-view aggregators for one-shot radiance field reconstruction. Building on these cross-scene Nerf methods, our work introduces a generalized semantic and rendering joint field, aiming to achieve simultaneous cross-scene reconstruction and segmentation.

2.2. NeRFs with Scene Understanding

Encoding semantics into NeRF is essential for scene understanding. Semantic NeRF [54] first explored introducing vanilla NeRF into semantic masks by adopting extra MLP layers to "render" semantic labels. DFF [18] and FeatureNeRF [47] utilize the pre-trained CLIP network and employ extra MLP layers for distillation learning to render text-aligned semantics features. Panoptic-Lifting [34] directly distills labels from Mask2former's predicted probabilities [7]. Based on Generalize NeRF [27], Semantic-Ray [22] adds an additional semantic branch to perform per-pixel semantic label rendering.

In conclusion, although these methods have extended the idea, *e.g.*, by applying to panoptic tasks [13], adding large language model (LLM) [32] features [4, 18, 47], and making it generalize [22], they all consider the semantic prob-

lem as another "rendering" variant: they render labels or features for each pixel independently, ignoring the contextual consistency among pixels in the novel view.

In contrast to previous approaches, we frame the segmentation issue as "prediction with context" rather than "isolated label rendering". Accordingly, we generate semantic-aware features instead of labels from our semantic-embedding field in new views. Moreover, we are able to perform context-aware segmentation thanks to the capabilities of the segmenter, which is a feature that previous methods lacked. Thanks to this design, the rendering and segmentation branches can benefit each other. Therefore, unlike [16], which enhances 3D object detection performance at the expense of reconstruction performance, our method can simultaneously improve both reconstruction and segmentation performance.

3. Preliminaries

In this section, we take a brief review of GNT [38]. NeRF represents a 3D scene as a radiance field $\mathcal{F} : (\boldsymbol{x}, \boldsymbol{\theta}) \mapsto (\mathbf{c}, \sigma)$, which maps the spatial coordinate $\mathbf{x}$ to a density σ and color $\mathbf{c}$. While GNT models 3D scene as a coordinate-aligned feature field $\mathcal{F} : (\boldsymbol{x}, \boldsymbol{\theta}) \mapsto \boldsymbol{f} \in \mathbb{R}^d$, d is the dimension of the features. To learn this representation, GNT uses Transformer as a set aggregated function $\mathcal{V}(\cdot)$ to aggregate the features of reference views into a coordinate-aligned feature field, which is formulated below:

$$\mathcal{F}(\boldsymbol{x}, \boldsymbol{\theta}) = \mathcal{V}(\boldsymbol{x}, \boldsymbol{\theta}; \{\boldsymbol{I}_1, \cdots, \boldsymbol{I}_N\}) \qquad (1)$$

Subsequently, to obtain the final outputs C of the ray $r = (o, d)$ in target view in this feature field, GNT parameterizes the ray by $r(t) = o + td$, $t \in [t_1, t_M]$, and uniformly samples M points x_i of feature representations $f_i = \mathcal{F}(x_i, \theta) \in \mathbb{R}^d$ along the ray r. Then, GNT adopts Transformer as a formulation of weighted aggregation to achieve volume rendering:

$$C(r) = \text{MLP} \circ \frac{1}{M} \sum_{i=1}^{M} A(x_i) f(x_i, \theta), \qquad (2)$$

where $A(x_i)$ is the weight of point x_i computed by Transformer and $C(r)$ is the rendered color of the ray r.

4. Methodology

4.1. Overall Framework

Given N images $I = \{I_i \in \mathbb{R}^{H \times W \times 3}\}$ with corresponding poses, the training targets are to conduct scene perception (semantic Y_{sem}, instance Y_{ins}) and reconstruction Y_{rgb} in the novel *target* views, where $Y_{sem} = \{Y_i \in \mathbb{R}^{H \times W \times O}\}$, $Y_{ins} = \{Y_i \in \mathbb{R}^{H \times W \times C}\}$, and $Y_{rgb} = \{Y_i \in \mathbb{R}^{H \times W \times 3}\}$, where O and C denote the number of semantic classes and instances. Unlike previous Semantic NeRF methods that directly render colors and semantic labels in a per-pixel manner, we perform segmentation tasks using (implicit) image context (Fig. 2). To accomplish this objective, we utilize NeRF to aggregate novel view semantic features S_{sem}^{2D} from reference features F_i^{sem}(Sec. 4.2), where F_i^{sem} is extracted by Multi-Scale Feature Extractor. After that, semantic features S_{sem}^{2D} are fed into the Context-Aware Perception Head to perform image-wise context-aware perception.

Multi-Scale Feature Extractor. To enhance the semantic-aware of our semantic-embedding fields, for each reference image I_i, we use shared ResNet-34 followed by a Feature Pyramid Network (FPN) [21] module to produce multi-scale features F_i^{sem} for our semantic field aggregation.

Context-Aware Perception Head. Our perception head takes rendered semantic features S_{sem}^{2D} and outputs semantic labels Y_{sem} in novel view. Here we split S_{sem}^{2D} into 4 parts $[s_{sem,1}^{2D}, s_{sem,2}^{2D}, s_{sem,3}^{2D}, s_{sem,4}^{2D}]$ to decompose high-level features and low-level features, and adopt the decoder of the U-Net [33] to verify our architecture's performance. In specific, for i-th layer, it consists of an upsampling (i-1)-th layer's output feature s'_{i-1} with 2×2 convolution("up-convolution"), a concatenation of i-th feature map $s_{sem,i}^{2D}$, and two 3×3 convolutions followed by a ReLU. The process can be formulated as below:

$$s'_i = \text{ReLU} \cdot \text{Conv}(s_{sem,i}^{2D} + \text{Up-Conv}(s'_{i-1})) \qquad (3)$$

Rendering and Training Process. NeRF can only render limited N_{pts} points in each iteration, the same as our method. During rendering, we stack all the semantic features $S_{sem}^{2D}(r)$ of sampled points as image-level features
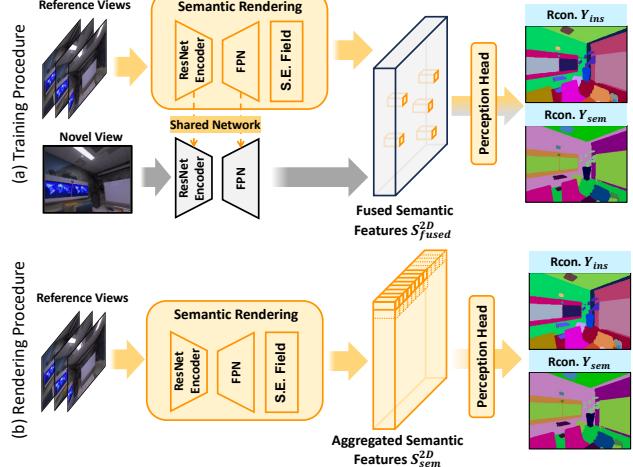

Figure 3. Illustration of **training**(a) and **rendering**(b) procedure, where S.E. field denotes Semantic-Embedding Field.

and feed them into the perception head together (see Fig. 3(b)). However, it's impossible to use fully rendered semantic features in every training batch. Therefore, as shown in Fig. 3(a), for the semantic 2D map S_{sem}^{2D}, we specifically fill its unrendered areas with the corresponding regions from the novel 2D map S_{novel}^{2D}. This process creates a fused image-level feature map S_{fused}^{2D}, which is subsequently fed into the Perception Head for semantic prediction.

4.2. Co-Aggregated Fields and Joint Rendering

Given low-level features F_i^{rgb} and high-level features F_i^{sem} from Multi-Scale Feature Extractor, we use shared-attention(*i.e.* Field-Aggregation Transformer) to co-aggregate the radiance field and semantic-embedding field. Subsequently, another shared-attention (*i.e.* Ray-Aggregation Transformer) employs joint volumetric rendering from both fields to generate point-wise colors and semantic features in the novel view.

Co-Aggregate Radiance and Semantic-Embedding Fields. We represent a 3D scene as a coordinate-aligned feature field [38], which can attach low-level features for ray rendering or high-level features for scene understanding. Therefore, to obtain feature representations of position x in novel view, following the idea of epipolar geometry constraint [36], x is projected to every reference image and interpolated the feature vector on the image plane. Firstly, the Field Aggregation Transformer (dubbed FAT($\cdot$)) is adopted to combine all features F_i^{rgb} from reference views for radiance field $\mathcal{F}^{rgb}(x, \theta)$ aggregation. Formally, this process can be written as:

$$\mathcal{F}^{rgb}(x, \theta), \mathcal{A}_{FAT} = \text{FAT}(F_1^{rgb}(\Pi_1(x), \theta), \cdots,$$
$$F_N^{rgb}(\Pi_N(x), \theta)), \qquad (4)$$

where $\Pi_i(x)$ projects x to i-th reference image plane by applying extrinsic matrix, $F_i^{rgb}(\Pi_i(x), \theta) \in \mathbb{R}^{D_{rgb}}$ com-

putes the feature vector at projected position $\Pi_i(\boldsymbol{x}) \in \mathbb{R}^2$ via bilinear interpolation on the feature grids. Furthermore, $\mathcal{A}_{FAT} \in \mathbb{R}^{N_{pts} \times N}$ is the aggregation weight from Field Aggregation Transformer, which enables us to construct semantic embedding field $\mathcal{F}^{sem}(\boldsymbol{x}, \boldsymbol{\theta})$ easily by applying dot-product with features $\boldsymbol{F}_i^{sem}$ from reference views:

$$\mathcal{F}^{sem}(\boldsymbol{x}, \boldsymbol{\theta}) = \text{Mean} \circ (\mathcal{A}_{FAT} \cdot [\boldsymbol{F}_1^{sem}(\Pi_1(\boldsymbol{x}), \boldsymbol{\theta}), \\ \cdots, \boldsymbol{F}_N^{sem}(\Pi_N(\boldsymbol{x}), \boldsymbol{\theta})]^T) \quad (5)$$

The network detail of the Field-Aggregation Transformer can refer to the appendix.

Joint Volumetric Rendering from both Fields. For radiance rendering, given a sequence of $\{\boldsymbol{f}_1^{rgb}, \cdots, \boldsymbol{f}_M^{rgb}\}$ from a sample ray, where $\boldsymbol{f}_i^{rgb} = \mathcal{F}^{rgb}(\boldsymbol{x}_i, \boldsymbol{\theta}) \in \mathbb{R}^{D_{rgb}}$ is the radiance feature of sampled points $\boldsymbol{x}_i$ along its corresponding sample ray $\boldsymbol{r} = (\boldsymbol{o}, \boldsymbol{d})$, we apply Ray-Aggregation Transformer (dubbed $\text{RAT}(\cdot)$) to aggregate weighted attention $\mathcal{A}_{RAT} \in \mathbb{R}^{N_{pts}}$ of the sequence to assemble the final feature vectors $S_{rgb}^{2D} \in \mathbb{R}^{D_{rgb}}$, then mean pooling and MLP layers are employed to map the feature vectors to RGB. The formulation of the above process is written below:

$$S_{rgb}^{2D}(\boldsymbol{r}), \mathcal{A}_{RAT} = \text{RAT}(\boldsymbol{f}_1^{rgb}, \cdots, \boldsymbol{f}_M^{rgb}) \\ C(\boldsymbol{r}) = \text{MLP} \circ \text{Mean} \circ S_{rgb}^{2D}(\boldsymbol{r}) \quad (6)$$

For semantic rendering, similar to the process of co-aggregate fields, given a sequence of $\{\boldsymbol{f}_1^{sem}, \cdots, \boldsymbol{f}_M^{sem}\}$ from the same sampled ray, we adopt dot-product between $\mathcal{A}_{RAT}$ and $\boldsymbol{f}_i^{sem} \in \mathbb{R}^{D_{sem}}$ to render semantic features $S_{sem}^{2D}(\boldsymbol{r}) \in \mathbb{R}^{D_{sem}}$ in novel view:

$$S_{sem}^{2D}(\boldsymbol{r}) = \text{MLP} \circ \text{Mean} \circ (\mathcal{A}_{RAT} \cdot [\boldsymbol{f}_1^{sem}, \cdots, \boldsymbol{f}_M^{sem}]^T) \quad (7)$$

The network detail of the Ray-Aggregation Transformer can be referred to the appendix.

4.3. Optimizations

We train the whole network from scratch under photometric loss $\mathcal{L}_{rgb}$, semantic pixel loss $\mathcal{L}_{sem}$ as well as our proposed semantic distill loss $\mathcal{L}_{disill}^{2D}$ and depth-guided semantic distill loss $\mathcal{L}_{distill}^{dgs}$, the overall loss $\mathcal{L}_{all}$ can be summarized as:

$$\mathcal{L}_{all} = \alpha_1 \cdot \mathcal{L}_{rgb} + \alpha_2 \cdot \mathcal{L}_{sem} + \alpha_3 \cdot \mathcal{L}_{distill}^{2D} + \alpha_4 \cdot \mathcal{L}_{distill}^{dgs} \quad (8)$$

Photometric loss $\mathcal{L}_{rgb}$ and semantic pixel loss $\mathcal{L}_{sem}$ are pixel-level supervision, and they are widely used in NeRF and semantic tasks:

$$\mathcal{L}_{rgb} = \sum_{\mathbf{r} \in \mathcal{R}} \left\| \hat{\mathbf{C}}(\mathbf{r}) - \mathbf{C}(\mathbf{r}) \right\|_2^2, \quad (9)$$

$$\mathcal{L}_{sem} = -\sum_{\mathbf{r} \in \mathcal{R}} \left[\sum_{l=1}^C p^c(\mathbf{r}) \log \hat{p}^c(\mathbf{r}) \right], \quad (10)$$
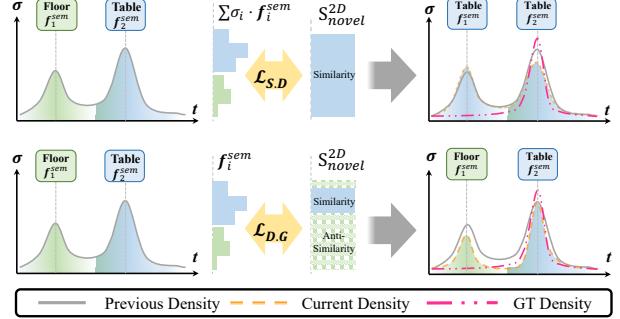

Figure 4. **2D Semantic Distillation** $\mathcal{L}_{\text{S.D}}$ and **Depth-Guided Semantic Optimization** $\mathcal{L}_{\text{D.G}}$. This figure demonstrates a single raw of our semantic-embedding field. the network "cheat" by rendering all points $\boldsymbol{f}_i^{sem}$ to the same prediction to satisfy $\mathcal{L}_{\text{S.D}}$ supervision. By performing spatial-wise semantic supervision, $\mathcal{L}_{\text{S.D}}$ is able to mitigate the issue of "cheating".

where $\mathcal{R}$ are the sampled rays within a training batch. $\hat{\mathbf{C}}(\mathbf{r}), \mathbf{C}(\mathbf{r})$ are the GT color and predicted color for ray r, respectively. Moreover, p^c and $\hat{p}^c$ are the multi-class semantic probability at class c of the ground truth map.

2D Semantic Distillation. For semantic-driven tasks, it is crucial to augment the discrimination and semantic-aware ability of our rendered features. Therefore, we propose **2D Semantic Distill Loss** $\mathcal{L}_{\text{S.D}}$. It distills [15] the aggregated features $\boldsymbol{S}_{sem}^{2D}$ by considering the features $\boldsymbol{S}_{novel}^{2D}$ extracted on novel-view as teacher, which effectively minimizes the differences between aggregated and teacher features:

$$\mathcal{L}_{\text{S.D}} = \sum_{\mathbf{r} \in \mathcal{R}} \left[1 - \cos \left(\boldsymbol{S}_{sem}^{2D}(r), \boldsymbol{S}_{novel}^{2D}(r) \right) \right] \quad (11)$$

Since our model is trained from scratch, we apply a **gradient block** after ResNet-34 encoder to ensure that the loss function supervises the aggregation process of the Transformer modules to get better rendered semantic features $\boldsymbol{S}_{sem}^{2D}$, otherwise, the extractor tends to learn less discriminative features to "cheat" the distillation loss.

Depth-Guided Semantic Optimization. It's worth noting that although $\mathcal{L}_{\text{S.D}}$ it significantly boosts the discrimination of rendered features, it also corrupts the geometry representation of our model. As illustrated in the first column of Fig. 4, the semantic representation of the ray is conducted by weighted summation of sampled point $\boldsymbol{f}_i^{sem}$ and their corresponding coefficient σ_i, where σ_i belongs to $\mathcal{A}_{RAT}$. Therefore, the loss can be minimized by misguiding $\boldsymbol{f}_i^{sem}$ (class 'Floor'$\rightarrow$'Table') rather than optimizing the attention weights σ_i(i.e. geometry representation). To restore the semantic consistency with geometry constraint, we proposed Depth-Guided Semantic Optimization $\mathcal{L}_{\text{D.G}}$. given a sequence of sampled points $\boldsymbol{x}_i$ and corresponding features $\boldsymbol{f}_i^{sem}$ from ray $\mathbf{r}$, we perform per-point semantic distillation from the teacher's features $S_{novel}^{2D}(\boldsymbol{r})$:

$$\mathcal{L}_{\text{D.G}} = \sum_{\mathbf{r} \in \mathcal{R}} \sum_{i=1}^{N_{pts}} L_{sim}(x_i, \boldsymbol{f}_i^{sem}, \boldsymbol{S}_{novel}^{2D}(\mathbf{r})) \quad (12)$$

Method	Settings	Synthetic Data (Replica [35])			Real Data (ScanNet [10])		
		Total Acc↑	Avg Acc↑	mIoU↑	Total Acc↑	Avg Acc↑	mIoU↑
MVSNeRF + Semantic Head	Generalization	54.25	33.70	23.41	60.01	46.01	39.82
NeuRay + Semantic Head		69.35	43.97	35.90	77.61	57.12	51.03
Semantic-Ray		70.51	47.19	41.59	78.24	62.55	57.15
Ours		**78.01**	**50.80**	**48.53**$_{6.94↑}$	**78.49**	**70.75**	**59.92**$_{2.7↑}$
Semantic-NeRF	Finetuning	94.36	70.20	75.06	97.54	93.89	91.24
MVSNeRF + Semantic Head$_{ft}$		79.48	62.85	53.77	76.25	69.70	55.26
NeuRay + Semantic Head$_{ft}$		85.54	70.05	63.73	91.56	81.04	77.48
S-Ray$_{ft}$		96.38	80.81	75.96	98.20	93.97	91.06
Ours$_{ft}$		**97.60**	**86.45**	**87.72**$_{11.76↑}$	**98.43**	**94.77**	**93.84**$_{2.78↑}$

Table 1. Quantitative Comparison with other SOTA methods for generalized and fine-tuning semantic segmentation.
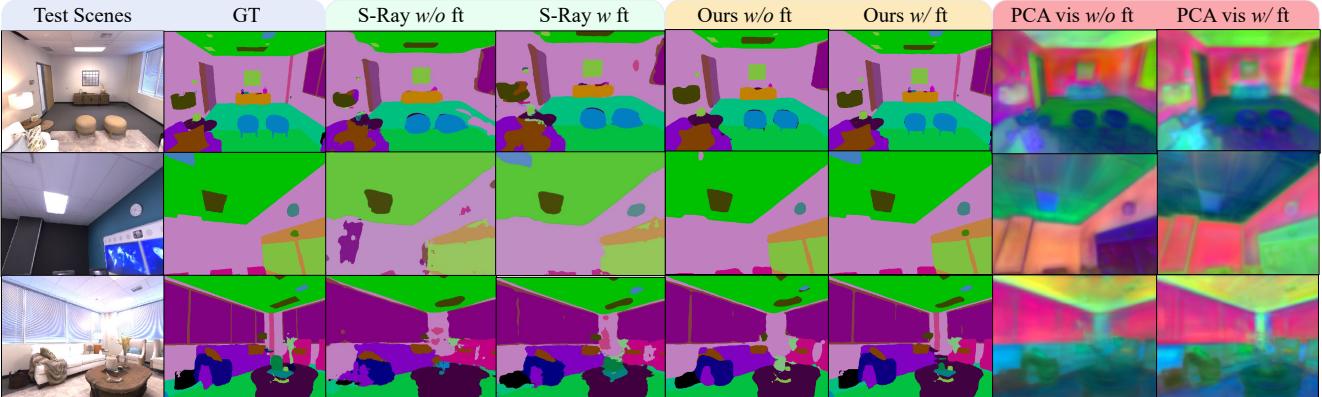

Figure 5. **Semantic quality comparison in Replica** [35]. On the left, we show the rendering results of S-Ray [22] and GP-NeRF(ours) in generalized and finetuning settings. On the right, we visualize the PCA results of our rendered semantic features in novel views.

where L_{sim} is the cosine embedding loss, it performs supervision under two situations: (1) for those points x_i near the GT depth ($|x_i - x_d| < N_p$), it conducts similarity constraint with teacher features; (2) for those points far from the GT depth ($|x_i - x_d| > N_p$), it conducts anti-similarity constraint with teacher features, where x_d is the sampled point projected by GT depth. In our implementation, N_p is set to 2. The formulation is shown below:

$$L_{sim}(x_i, f_1, f_2) = \begin{cases} 1 - \cos(f_1, f_2) & , |x_i - x_d| < N_p \\ \max(0, \cos(f_1, f_2)) & , |x_i - x_d| > N_p \end{cases}$$
(13)

5. Experiments

5.1. Implementation Details

We conduct experiments to compare our method against state-of-the-art methods for novel view synthesis with RGBs as well as semantic/instance labels. Firstly, we train our model in several scenes and directly evaluate our model on test scenes (i.e., unseen scenes). Secondly, we finetune our generalized model on each unseen scene with small steps and compared them with per-scene optimized NeRF methods in semantic and reconstruction metrics.

Parameter Settings. We train our method end-to-end on datasets of multi-view posed images using the Adam opti-

mizer to minimize the overall loss $\mathcal{L}_{all}$. The learning rate or Multi-Task Feature Extractor, Transformer modules, and Perception Head are 5×10^{-3}, 1×10^{-5} and 5×10^{-5} respectively, which decay exponentially over training steps. For generalized training, we train for 200,000 steps with 512 rays sampled in each iteration. For finetuning, we train for 10,000 steps for each scene. Meanwhile, we sample 64 points per ray across all experiments. For each render interaction, we select $N = 10$ images as reference views.

Metrics. Same as Semantic-Ray [22]: (1) For semantic quality evaluation, we adopt mean Intersection-over-Union (mIoU) as well as average accuracy and total accuracy to compute segmentation quality. (2) For render quality evaluation, Peak Signal-to-Noise Ratio (PSNR), Structural Similarity Index Measure (SSIM) [42], and the Learned Perceptual Image Patch Similarity (LPIPS) [52] are adopted. More specifically, we refer to DM-NeRF [40] and use AP of all 2D test images to evaluate instance quality evaluation.

Datasets. We train and evaluate our method on Replica [35] and ScanNet [10] datasets. In these experiments, we use the same resolution and train/test splits as S-Ray [22].

5.2. Comparison with State-of-the-Art

Generalized Semantic Results. We compare our model with Semantic Ray, Generalized NeRFs(*i.e.* NeuRay, MVS-

Scene	M-RCNN	Swin-T	DM-NeRF	Ours
Office_0	74.05	80.17	82.71	90.46
Office_2	73.41	75.39	81.12	88.62
Office_3	72.91	73.26	76.30	82.64
Office_4	74.76	72.51	70.33	88.38
Room_0	78.67	76.90	79.83	89.91
Room_1	78.38	81.41	92.11	93.38
Room_2	77.58	80.33	84.78	93.11
Average	75.68	77.14	81.03	$89.50_{8.47\uparrow}$

Table 2. Quantitative results of instance segmentation results on Replica [35]. The metric is $AP^{0.75}$.

NeRF) with Semantic Head, and classical semantic segmentor (SemanticFPN) in both synthesis [35] and real-world [10] datasets. We render the novel images in the resolution of 640×480 for Replica, and 320×240 for ScanNet. As shown in Tab. 1, our method achieves remarkable performance improvements compared with baselines. For example, our method significantly improves over Semantic-Ray by 6.94% in Replica and 2.7% in ScanNet. It's notable that Replica has more categories than ScanNet, and we achieve higher performance improvements in Replica, which further demonstrates the robustness and effectiveness of our semantic embedding field in handling complex semantic contexts.

Fine-tuning Semantic Results. We fine-tune our pre-trained with *10k* steps for per-scene optimize evaluation. In Tab. 1, we observe that our method is superior to not only generalized methods but also per-scene optimization methods. Especially in ScanNet evaluation, we outperform the per-scene optimized method Semantic-NeRF [54] by a notable margin of 2.6% in the mIoU metric. Comparatively, Semantic-Ray [22] performs 0.18% less effectively in the same metric. Furthermore, the visual results in Fig. 5 clearly reflect the quantitative results of Tab. 1. Given the benefit of jointly optimized attention maps to construct semantic embedding fields, our method demonstrates a clear ability to segment the boundaries of different classes effectively. This capability is particularly evident in the areas encircled in the figures.

Instance Segmentation Results. With the success of our method in semantic scene representation, we explore the potential of our method in instance-level decomposition. Given the reason that the objects of each scene are unique, we only evaluate our performance in the per-scene optimization setting. Tab. 2 presents the quantitative results. Not surprisingly, our method achieves excellent results for novel view prediction (+8.47% *w.r.t.* DM-NeRF [40]) thanks to our powerful semantic embedding field and context-aware ability in novel view prediction. Figures 6(a) further demonstrate that our semantic field can provide more discriminate semantic pattern than per-scene optimization method to decompose instances with
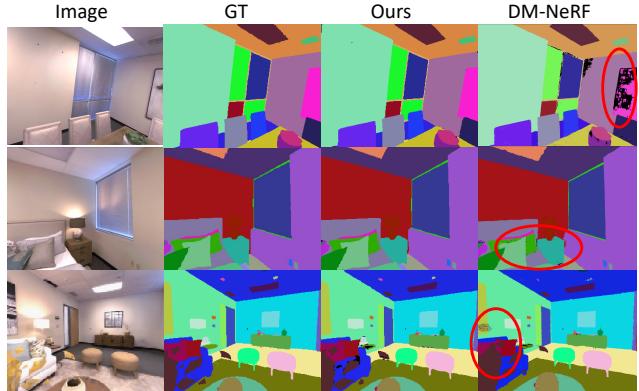

Figure 6. Visualization of instance segmentation results on synthesis dataset [35]. The discriminate area is highlighted with '◯'.
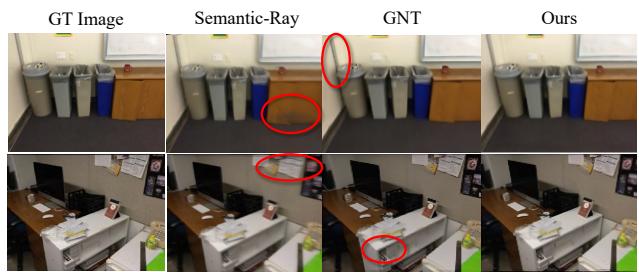

Figure 7. Qualitative results of scene rendering for generalization settings in ScanNet [10]. We plot the discriminate area with '◯'.

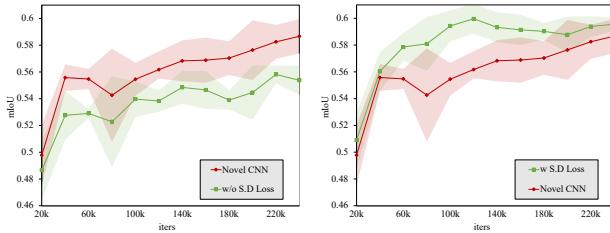| Method	PSNR↑	SSIM↑	LIPIPS↓
Semantic-NeRF	25.07	0.797	0.196
MVSNeRF	23.84	0.733	0.267
NeuRay	27.22	0.840	0.138
Semantic-Ray	26.57	0.832	0.173
Semantic-Ray$_{ft}$	29.27	0.865	0.127
GNT	28.96	0.909	0.135
GNT$_{ft}$	29.55	0.917	0.102
Ours	$\mathbf{29.37}_{2.8\uparrow}$	**0.919**	**0.110**
Ours$_{ft}$	$\mathbf{29.60}_{0.33\uparrow}$	**0.923**	**0.102**

Table 3. Reconstruction Quality in ScanNet [10]. '*ft*' denotes per-scene optimization using a generalized pre-trained model.

accurate boundaries. Moreover, our method prevents the mis-segmentation of pixels within an instance thanks to our context-aware ability. These features enhance the accuracy and reliability of our scene perception process.

Reconstruction Results. It's worth noting that our method not only achieves SOTA in perception evaluation but also surpasses other SOTA methods in reconstruction quality. As shown in Tab 3, in the generalized setting, our method surpasses Semantic-Ray [22] by 2.8% in PSNR, which is even better than Semantic-Ray with fine-tuning steps. Subsequently, we also improve the reconstruction quality by 0.41% compared with GNT [38] given the benefit on our radiance field is also supervised from semantic consistency. Fig. 7 provides visual evidence of our performance on ray

ID	Jointly Optimized	2D S.D Loss	Gradient Block	D.G Loss	mIoU↑	PSNR↑
1	✗	✗	✗	✗	56.45	29.06
2	✓	✗	✗	✗	57.19	29.30
3	✓	✓	✗	✗	52.03	29.29
4	✓	✓	✓	✗	59.55	29.26
5	✓	✓	✓	✓	59.92	29.37

Table 4. Ablations of our design choices on ScanNet [10]. Notice that 'Gradient Block' is dependent on '2D S.D Loss' and 'D.G Loss', where 2D S.D denotes 2D Semantic Distill Loss and D.G denotes Depth-Guided Semantic Enhancement.

(a) *w/o* S.D Loss and gradient block. (b) *w/* S.D Loss and gradient block.

Figure 8. Ablations of **Semantic Distillation Loss via Gradient Block**. Red part denotes the mIoU results predicted by extracted features from novel image. Green part denotes the mIoU predicted by the rendered features from semantic embedding fields.

rendering reconstruction, where our method delivers more detailed and clearer reconstruction results.

5.3. Component Analysis and Ablation Study

Jointly Optimized Attention Maps. As illustrated in sec. 4.2, we aggregate semantic-embedding fields and render semantic features in novel views by sharing attention maps from Transformer modules. In Tab. 4, we compare the influence of our jointly optimized Field in ID. 1, 2, and evaluate their scene perception and reconstruction performances. In experiment ID. 1, when constructing the semantic field and aggregating features in novel views, we freeze the Attention maps from Transformers. Conversely, in experiment ID. 2, we unfreeze the attention maps and jointly optimize them through semantic and radiance supervision. Obviously, joint optimization can achieve better performance in semantic perception and ray reconstruction by 0.74% and 0.24%, compared with the frozen patterns. This approach further demonstrates that semantic consistency can provide a radiance reference for pixels within the same classes. Additionally, radiance consistency also contributes to achieving more accurate boundary segmentation.

Semantic Distill Loss and Gradient Block. ID. 3, 4 in Tab. 4 reflect the influence of 2D semantic distillation loss and corresponding gradient block. As observed, there is a significant drop in performance (-5.16 compared to ID. 2) when only the 2D semantic distillation loss is adopted, which means the shared parts of the teacher and student branch (*i.e.* CNN encoder and FPN) tend to learn

less discriminate features to "cheat" the distillation loss. Meanwhile, with our **Gradient Block**, the situation can be solved, and the performance of mIoU achieves remarkable improvements by 7.52%. Moreover, we repeat the ID.3, 4 experiments five times and show the mIoU learning curves on ScanNet [10] in Fig. 8. We can observe that this contribution leads to a more precise convergence speed and higher final accuracy (See Fig. 8(b)).

Depth-Guided Semantic Distill Loss. It is notable that 2D semantic distill has a negative impact on reconstruction quality, by 0.4% in PSNR compared with ID. 2, which is due to the fact that the 2D semantic distill loss can only supervise the rendered features rather than 3D points within the rays. Under this circumstance, some points in the ray would be "cheated" by adjusting the semantic representation to satisfy distillation loss, which would further impact the actual weight distribution of the points in sample rays. ID. 5 in Fig. 4 shows that $\mathcal{L}_{D.G}$ yields clear improvement by 0.37% and 0.11% in mIoU and PSNR, indicating that a more precise, 3D-level semantic supervision can partially improve the geometry awareness of our semantic field and suppress the "cheating" phenomenon.

6. Conclusion

In this paper, we propose GP-NeRF, the first unified learning framework that combines NeRF and segmentation modules to perform context-aware 3D scene perception. Unlike previous NeRF-based approaches that render semantic labels for each pixel individually, the proposed GP-NeRF utilizes many contextual modeling units from the widely-studied 2D segmentors and introduces Transformers to co-construct radiance as well as semantic embedding fields and facilitates the joint volumetric rendering upon both fields for novel views. New self-distillation mechanisms are also designed to further boost the quality of the semantic embedding field. Comprehensive experiments demonstrate that GP-NeRF achieves significant performance improvements (sometimes $> 10\%$) compared to existing SOTA methods. In the future, we will follow more recent studies on visual saliency [14, 23, 37] as well as semi-supervised learning techniques [5, 25, 45, 51] to overcome the scenario of lack of full annotated semantic labels [9, 19, 20, 50] in scene understanding task.

7. Acknowledgement

References

[1] Muhammad Awais, Muzammal Naseer, Salman Khan, Rao Muhammad Anwer, Hisham Cholakkal, Mubarak Shah, Ming-Hsuan Yang, and Fahad Shahbaz Khan. Foundational models defining a new era in vision: A survey and outlook. *arXiv preprint arXiv:2307.13721*, 2023. 2

[2] 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 *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5855–5864, 2021. 2

[3] Jonathan T Barron, Ben Mildenhall, Dor Verbin, Pratul P Srinivasan, and Peter Hedman. Mip-nerf 360: Unbounded anti-aliased neural radiance fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5470–5479, 2022. 2

[4] Jiazhong Cen, Zanwei Zhou, Jiemin Fang, Wei Shen, Lingxi Xie, Xiaopeng Zhang, and Qi Tian. Segment anything in 3d with nerfs. *arXiv preprint arXiv:2304.12308*, 2023. 3

[5] Changrui Chen, Jungong Han, and Kurt Debattista. Virtual category learning: A semi-supervised learning method for dense prediction with extremely limited labels. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024. 8

[6] Hao Chen, Bo He, Hanyu Wang, Yixuan Ren, Ser Nam Lim, and Abhinav Shrivastava. Nerv: Neural representations for videos. *Advances in Neural Information Processing Systems*, 34:21557–21568, 2021. 3

[7] Bowen Cheng, Ishan Misra, Alexander G Schwing, Alexander Kirillov, and Rohit Girdhar. Masked-attention mask transformer for universal image segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1290–1299, 2022. 2, 3

[8] Bowen Cheng, Alex Schwing, and Alexander Kirillov. Per-pixel classification is not all you need for semantic segmentation. *Advances in Neural Information Processing Systems*, 34:17864–17875, 2021. 2

[9] De Cheng, Gerong Wang, Bo Wang, Qiang Zhang, Jungong Han, and Dingwen Zhang. Hybrid routing transformer for zero-shot learning. *Pattern Recognition*, 137:109270, 2023. 8

[10] Angela Dai, Angel X Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5828–5839, 2017. 1, 2, 6, 7, 8, 12, 13

[11] Kangle Deng, Andrew Liu, Jun-Yan Zhu, and Deva Ramanan. Depth-supervised nerf: Fewer views and faster training for free. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12882–12891, 2022. 3

[12] Di Feng, Christian Haase-Schütz, Lars Rosenbaum, Heinz Hertlein, Claudius Glaeser, Fabian Timm, Werner Wiesbeck, and Klaus Dietmayer. Deep multi-modal object detection and semantic segmentation for autonomous driving: Datasets, methods, and challenges. *IEEE Transactions on Intelligent Transportation Systems*, 22(3):1341–1360, 2020. 2

[13] Xiao Fu, Shangzhan Zhang, Tianrun Chen, Yichong Lu, Lanyun Zhu, Xiaowei Zhou, Andreas Geiger, and Yiyi Liao. Panoptic nerf: 3d-to-2d label transfer for panoptic urban scene segmentation. In *2022 International Conference on 3D Vision (3DV)*, pages 1–11. IEEE, 2022. 2, 3

[14] Guangyu Guo, Longfei Han, Le Wang, Dingwen Zhang, and Junwei Han. Semantic-aware knowledge distillation with parameter-free feature uniformization. *Visual Intelligence*, 1(1):6, 2023. 8

[15] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015. 5

[16] Benran Hu, Junkai Huang, Yichen Liu, Yu-Wing Tai, and Chi-Keung Tang. Nerf-rpn: A general framework for object detection in nerfs. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 23528–23538, 2023. 3

[17] Maximilian Jaritz, Jiayuan Gu, and Hao Su. Multi-view pointnet for 3d scene understanding. In *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*, pages 0–0, 2019. 2

[18] Sosuke Kobayashi, Eiichi Matsumoto, and Vincent Sitzmann. Decomposing nerf for editing via feature field distillation. *Advances in Neural Information Processing Systems*, 35:23311–23330, 2022. 3

[19] Hao Li, Dingwen Zhang, Nian Liu, Lechao Cheng, Yalun Dai, Chao Zhang, Xinggang Wang, and Junwei Han. Boosting low-data instance segmentation by unsupervised pretraining with saliency prompt. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15485–15494, 2023. 8

[20] Chao Liang, Fan Ma, Linchao Zhu, Yingying Deng, and Yi Yang. Caphuman: Capture your moments in parallel universes. *arXiv preprint arXiv:2402.00627*, 2024. 8

[21] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2117–2125, 2017. 4

[22] Fangfu Liu, Chubin Zhang, Yu Zheng, and Yueqi Duan. Semantic ray: Learning a generalizable semantic field with cross-reprojection attention. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17386–17396, 2023. 2, 3, 6, 7, 13

[23] Nian Liu, Junwei Han, and Ming-Hsuan Yang. Picanet: Pixel-wise contextual attention learning for accurate saliency detection. *IEEE TIP*, 29:6438–6451, 2020. 8

[24] Nian Liu, Kepan Nan, Wangbo Zhao, Yuanwei Liu, Xiwen Yao, Salman Khan, Hisham Cholakkal, Rao Muhammad Anwer, Junwei Han, and Fahad Shahbaz Khan. Multi-grained temporal prototype learning for few-shot video object segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 18862–18871, 2023. 2

[25] Yuanwei Liu, Nian Liu, Qinglong Cao, Xiwen Yao, Junwei Han, and Ling Shao. Learning non-target knowledge for few-shot semantic segmentation. In *Proceedings of*

the IEEE/CVF conference on computer vision and pattern recognition, pages 11573–11582, 2022. 8

[26] Yuanwei Liu, Nian Liu, Xiwen Yao, and Junwei Han. Intermediate prototype mining transformer for few-shot semantic segmentation. *Advances in Neural Information Processing Systems*, 35:38020–38031, 2022. 2

[27] Yuan Liu, Sida Peng, Lingjie Liu, Qianqian Wang, Peng Wang, Christian Theobalt, Xiaowei Zhou, and Wenping Wang. Neural rays for occlusion-aware image-based rendering. In *CVPR*, 2022. 2, 3, 13

[28] 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. *Communications of the ACM*, 65(1):99–106, 2021. 2

[29] Michael Oechsle, Songyou Peng, and Andreas Geiger. Unisurf: Unifying neural implicit surfaces and radiance fields for multi-view reconstruction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5589–5599, 2021. 3

[30] Keunhong Park, Utkarsh Sinha, Jonathan T Barron, Sofien Bouaziz, Dan B Goldman, Steven M Seitz, and Ricardo Martin-Brualla. Nerfies: Deformable neural radiance fields. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5865–5874, 2021. 3

[31] Albert Pumarola, Enric Corona, Gerard Pons-Moll, and Francesc Moreno-Noguer. D-nerf: Neural radiance fields for dynamic scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10318–10327, 2021. 3

[32] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, pages 8748–8763. PMLR, 2021. 3

[33] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18*, pages 234–241. Springer, 2015. 4

[34] Yawar Siddiqui, Lorenzo Porzi, Samuel Rota Bulò, Norman Müller, Matthias Nießner, Angela Dai, and Peter Kontschieder. Panoptic lifting for 3d scene understanding with neural fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9043–9052, 2023. 2, 3

[35] Julian Straub, Thomas Whelan, Lingni Ma, Yufan Chen, Erik Wijmans, Simon Green, Jakob J Engel, Raul Mur-Artal, Carl Ren, Shobhit Verma, et al. The replica dataset: A digital replica of indoor spaces. *arXiv preprint arXiv:1906.05797*, 2019. 1, 6, 7, 12

[36] Mohammed Suhail, Carlos Esteves, Leonid Sigal, and Ameesh Makadia. Light field neural rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8269–8279, 2022. 4

[37] Yucheng Suo, Linchao Zhu, and Yi Yang. Text augmented spatial aware zero-shot referring image segmenta-

tion. In Houda Bouamor, Juan Pino, and Kalika Bali, editors, *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 1032–1043, Singapore, Dec. 2023. Association for Computational Linguistics. 8

[38] Mukund Varma T, Peihao Wang, Xuxi Chen, Tianlong Chen, Subhashini Venugopalan, and Zhangyang Wang. Is attention all that neRF needs? In *The Eleventh International Conference on Learning Representations*, 2023. 2, 3, 4, 7

[39] Dor Verbin, Peter Hedman, Ben Mildenhall, Todd Zickler, Jonathan T Barron, and Pratul P Srinivasan. Ref-nerf: Structured view-dependent appearance for neural radiance fields. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5481–5490. IEEE, 2022. 3

[40] Bing Wang, Lu Chen, and Bo Yang. Dm-nerf: 3d scene geometry decomposition and manipulation from 2d images. *arXiv preprint arXiv:2208.07227*, 2022. 2, 6, 7, 12

[41] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4690–4699, 2021. 2, 3

[42] 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. 6

[43] Suttisak Wizadwongsa, Pakkapon Phongthawee, Jiraphon Yenphraphai, and Supasorn Suwajanakorn. Nex: Real-time view synthesis with neural basis expansion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8534–8543, 2021. 2

[44] Dejia Xu, Yifan Jiang, Peihao Wang, Zhiwen Fan, Humphrey Shi, and Zhangyang Wang. Sinnerf: Training neural radiance fields on complex scenes from a single image. In *European Conference on Computer Vision*, pages 736–753. Springer, 2022. 3

[45] Wenhao Xue, Yang Yang, Lei Li, Zhongling Huang, Xinggang Wang, Junwei Han, and Dingwen Zhang. Weakly supervised point cloud segmentation via deep morphological semantic information embedding. *CAAI Transactions on Intelligence Technology*, 2023. 8

[46] Lior Yariv, Jiatao Gu, Yoni Kasten, and Yaron Lipman. Volume rendering of neural implicit surfaces. *Advances in Neural Information Processing Systems*, 34:4805–4815, 2021. 3

[47] Jianglong Ye, Naiyan Wang, and Xiaolong Wang. Featurenerf: Learning generalizable nerfs by distilling foundation models. *arXiv preprint arXiv:2303.12786*, 2023. 3

[48] Weicai Ye, Xinyue Lan, Shuo Chen, Yuhang Ming, Xingyuan Yu, Hujun Bao, Zhaopeng Cui, and Guofeng Zhang. Pvo: Panoptic visual odometry. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9579–9589, 2023. 2

[49] Alex Yu, Vickie Ye, Matthew Tancik, and Angjoo Kanazawa. pixelnerf: Neural radiance fields from one or few images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4578–4587, 2021. 3

[50] Dingwen Zhang, Guangyu Guo, Wenyuan Zeng, Lei Li, and Junwei Han. Generalized weakly supervised object localization. *IEEE Transactions on Neural Networks and Learning*

Systems, 2022. 8

[51] Dingwen Zhang, Hao Li, Wenyuan Zeng, Chaowei Fang, Lechao Cheng, Ming-Ming Cheng, and Junwei Han. Weakly supervised semantic segmentation via alternate self-dual teaching. *IEEE Transactions on Image Processing*, 2023. 8

[52] 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*, pages 586–595, 2018. 6

[53] Wenwei Zhang, Jiangmiao Pang, Kai Chen, and Chen Change Loy. K-net: Towards unified image segmentation. *Advances in Neural Information Processing Systems*, 34:10326–10338, 2021. 2

[54] 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*, pages 15838–15847, 2021. 2, 3, 7, 12, 13

GP-NeRF: Generalized Perception NeRF for Context-Aware 3D Scene Understanding

Supplementary Material

7.1. Implementation of our Transformers

We provide a simple and efficient pytorch pseudo-code to implement the attention operations in the field-aggregation, ray-aggregation transformer blocks in Alg. 1, 2. We use ray features to generate attention maps A_{field} and A_{ray}, and reuse them to construct semantic-embedding field as well as semantic features rendering.

Algorithm 1: Field-Aggregation Transformer

Input:
$X_0 \rightarrow$ coordinate aligned features (N_{rays}, N_{pts}, D_{rgb})
$X_{rgb} \rightarrow$ epipolar view Ray feats (N_{rays}, N_{pts}, N_{views}, D_{rgb})
$X_{sem} \rightarrow$ epipolar view Sem feats (N_{rays}, N_{pts}, N_{views}, D_{sem})

$\Delta d \rightarrow$ relative directions (N_{rays}, N_{pts}, N_{views}, 3)
Network: $f_Q, f_K, f_V, f_P, f_A, f_{rgb} \rightarrow$ MLP layers
Output: $S_{rgb}^{3D}, S_{sem}^{3D}$

Forward: Red for semantic-embedding field aggregation

1 $Q = f_Q\left(X_0\right), K = f_K\left(X_{rgb}\right), V = f_V\left(X_{rgb}\right)$
2 $P_{field} = f_P(\Delta d)$
3 $A_{field} = K - Q[:,:, \text{None}, :] + P$
4 $A_{field} = \text{softmax}(A, \dim = -2)$
5 $A'_{field} = A_{field} \cdot \text{repeat\_interleave}(4)$
6 $P'_{field} = P \cdot \text{repeat\_interleave}(4)$
7 $S_{rgb}^{3D} = ((V + P) \cdot A) \cdot \text{sum}(\dim = 2)$
8 $S_{rgb}^{3D} = f_{rgb}(S_{rgb}^{3D})$
9 $S_{sem}^{3D} = ((X_{sem} + P'_{field}) \cdot A'_{field}) \cdot \text{sum}(\dim = 2)$

7.2. Reconstruction results in instance setting

During the novel view instance segmentation task, we evaluate our reconstruction results and compare them with SOTA method DM-NeRF[40]. As shown in Table 5, our approach surpasses DM-NeRF in terms of SSIM and LPIPS metrics by 0.02% and 0.065%, respectively. It demonstrates that contextual information from semantic features can enhance the geometry reconstruction in our jointly optimized field and rendering framework.

7.3. Few-step Finetuning Comparison

Tab. 6 presents a comparison of different models, showcasing their mIoU and finetuning times on the ScanNet [10] dataset, along with the AP75 metric in Replica [35]. We observe that by finetuning with limited time, our model is able to achieve a better perception accuracy than a well-trained per-scene optimized method, such as 3.45% in

---

Algorithm 2: Ray-Aggregation Transformer

Input:
$X_0^{rgb} \rightarrow$ coordinate aligned rgb features (N_{rays}, N_{pts}, D_{rgb})
$X_0^{sem} \rightarrow$ coordinate aligned sem features (N_{rays}, N_{pts}, D_{sem})
$x \rightarrow$ point coordinates (after PE) (N_{rays}, N_{pts}, D_{rgb})
$d \rightarrow$ target view direction (after PE) (N_{rays}, N_{pts}, D_{rgb})
Network: $f_Q, f_K, f_V, f_P, f_A, f_{rgb}, f_{sem} \rightarrow$ MLP layers
Output: $S_{rgb}^{2D}, S_{sem}^{2D}$

Forward: Red for semantic-embedding field aggregation

1 $X_0^{rgb} = f_P(\text{concat}(X_0^{rgb}, d, x))$
2 $Q = f_Q\left(X_0^{rgb}\right), K = f_K\left(X_0^{rgb}\right), V = f_V\left(X_0^{rgb}\right)$
3 $A_{ray} = \text{matmul}\left(Q, K^T\right) / \sqrt{D}$
4 $A_{ray} = \text{softmax}(A_{ray}, \dim = -1)$
5 $A'_{ray} = A_{ray} \cdot \text{repeat\_interleave}(4)$
6 $S_{rgb}^{2D} = \text{matmul}(V, A_{ray})$
7 $S_{rgb}^{2D} = f_{rgb}(S_{rgb}^{2D})$
8 $S_{sem}^{2D} = \text{matmul}(X_0^{sem}, A'_{ray})$

Table 5. Quantitative results of reconstruction task in Replica[35] during instance segmentation setting.

Scene	DM-NeRF			Ours		
	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
Office_0	40.66	0.972	0.07	39.25	0.984	0.027
Office_2	36.98	0.964	0.115	36.01	0.974	0.042
Office_3	35.34	0.955	0.078	36.02	0.982	0.027
Office_4	32.95	0.921	0.172	32.75	0.94	0.085
Room_0	34.97	0.94	0.127	34.29	0.972	0.049
Room_1	34.72	0.931	0.134	36.45	0.968	0.043
Room_2	37.32	0.963	0.115	34.75	0.960	0.085
Average	36.13	0.949	0.116	35.64 0.49↓	0.969 0.02↑	0.051 0.065↓

mIoU with Semantic-NeRF [54] and 3.7% in AP75 with DM-NeRF [40]. Specifically, we observe that our method surpasses Semantic-Ray, requiring only half as many fine-tuning steps, and improves the mIoU by 0.74%, which further demonstrates that our semantic embedding field with more discrimination successfully improves the generalized ability.

We further evaluate the above experiments in instance segmentation setting, shown in the bottom column in Tab. 6. Not surprising, compared with SOTA method DM-NeRF[40], we achieve better performance with only 4k training steps, by 3.7% in AP75.

7.4. Additional Visualization Results

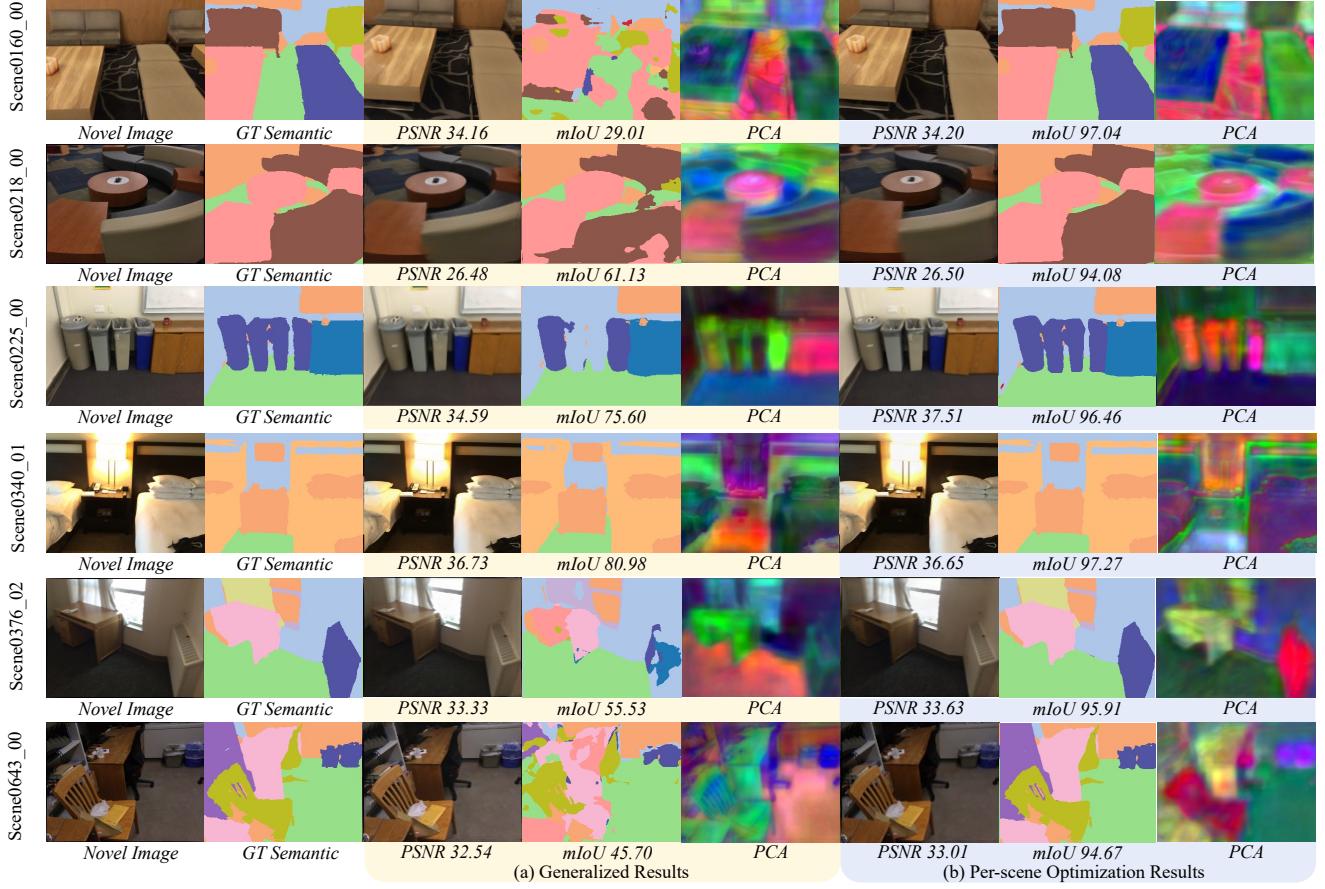Fig. 9 shows the additional qualitative results of semantic prediction and reconstruction.

Figure 9. The visualization results in ScanNet[10]. Here we visualize the semantic as well as reconstruction results in both generalized and finetuning settings.

Method	Train Step	Train Time	mIoU/AP75
Semantic-NeRF [54]	50k	~2h	89.33
MVSNeRF w/s-Ft	5k	~20min	52.02
NeuRay [27] w/s-Ft	5k	~32min	79.23
Semantic-Ray [22]-Ft	5k	~20min	92.04
Ours-Ft	2.5k	~20min	92.78 $_{0.74\uparrow}$
DM-NeRF	200k	~2h	81.03
Ours-Ft	4k	~30min	84.73 $_{3.7\uparrow}$

Table 6. mIoU and training steps/time on ScanNet [10]. "w/ s" means adding a semantic head on the baseline architectures.