A Additional Training and Architecture Details

A.1 Training details

We utilize the Adam optimizer with a learning rate of 0.0001, β_1 = 0.9, and β_2 = 0.999. Additionally, we implement learning rate warm-up for the initial 1,000 iterations. The minimum number K of objects in each scene is customized separately as follows: K = 8, 7, 5, 5, 2, and 5 for the CLEVR-567, CLEVR-3D, Room-Chair, Room-Diverse, LLFF, and MSN datasets, respectively.

To allow training on a high resolution, such as 256 × 256, we render individual pixels instead of large-sized patches. Specifically, we randomly sample a batch of 64 rays from the set of all pixels in the dataset, and then follow the hierarchical volume sampling to query 64 samples from the coarse network and 128 samples from the fine network.

In addition, we train our model from scratch, with the exception of the Room-Diverse dataset. The Room-Diverse dataset is more complex and requires an incremental learning approach. Specifically, we initialize our models for Room-Diverse using weights from a model that have previously been trained on the CLEVR-567 dataset.

A.2 Architecture details

Image encoder For the image encoder E, we utilize the well-known ResNet34 **?** architecture as the backbone, followed by three upsampling layers. Specifically, given the source image $\mathbf{I} \in \mathbb{R}^{256\times256\times3}$, we extract features $\mathbf{F}_1 \in \mathbb{R}^{128\times128\times64}$, $\mathbf{F}_2 \in \mathbb{R}^{64\times64\times128}$, and $\mathbf{F}_3 \in \mathbb{R}^{32\times32\times256}$ from the $conv2$, $conv3$ and $conv4$ layer of the ResNet34 architecture, respectively. Note that the max-pooling layer in $conv2$ is not used. Subsequently, these features are fed into the following three up-sampling layers within a U-net expansive path, resulting in the feature $\mathbf{F} \in \mathbb{R}^{128\times128\times512}$. The architecture of upsample layers is shown in Table 1.

Table 1: The architecture of upsample layers.All convolutional kernel sizes are 3×3. All activation functions for convolutional layers are ReLU. "+" indicates channel concatenation with a feature map of same resolution sourced from ResNet34.

Layer name	Input shape	Output shape	Stride
Conv1	$32 \times 32 \times 256$	$32 \times 32 \times 512$	1
Bilinear Upsampler	$32 \times 32 \times 512$	$64 \times 64 \times 512$	
Conv2	$64 \times 64 \times (512 + 128)$	$64 \times 64 \times 512$	1
Bilinear Upsampler	$64 \times 64 \times 512$	$128 \times 128 \times 512$	
Conv3	$128 \times 128 \times (512 + 64)$	$128 \times 128 \times 512$	1

Object NeRF To represent each object NeRF, we employ a simple 4-layer MLP that each layer has 128 channels and is followed by a ReLU activation. In addition, we use skip connection mechanism that add the first layer's activation to the third layer's activation. Note that the last layer outputs the RGB value with a sigmoid activation function.

Transformer-based module The efficient transformer module is a standard transformer decoder. Specifically, we firstly use each slot $\mathbf{z}_i$ as query and interact with other object slots with multi-head self-attention operation. Then we employ multi-head cross-attention operation to attend into and aggregate features from the flattend image features $E(\mathbf{I})$. Finally, we pass the resulting slot features into a feed forward network (FFN) to get the final slots. This transformer module is simple and easy to train than the slot attention module, as it does not contain a Gated Recurrent Unit (GRU) block.

Composition module The aggregation block performs a cross-attention operation, which aggregates object representations S with the 3D location x as the query to obtain the corresponding feature z. The attention block computes the similarity between S and z after mapping them into the same space through a linear layer, thus obtaining the probability distribution of x belonging to each slot. In the initial stages of our experiments, we observed that utilizing both modules simultaneously yielded superior results. We speculate that this improvement may be attributed to a good feature space alignment between 3D points and slot features using the proposed aggregation block.

B Object Segmentation on Real Images

Setup In this study, we use VIT-Base as the feature extractor for complex datasets of size 1008×756 with intricate textures. The adopted backbone networks are trained from scratch, and the method strictly follows unsupervised learning. To conserve memory, we resize the source image to 256× 256 and select only 7168 rays from the target view during each training iteration. Moreover, we define the task as foreground-background segmentation. As such, we set the number of slots to 2 and map it to two NeRFs. Each NeRF consists of a simple 4-layer MLP.

Unlike the setup on other datasets, we train sVORF on both LLFF scenes as two separate models Following the setup in NeRF-SOS, we divide the data of each scene into training and testing sets, ensuring there is no overlap between these two sets.

Results Figure 1 shows the complete segmentation results of sVORF on both *Flower* and *Fortress* datasets. The results demonstrate the potential of sVORF for 3D segmentation on non-object-centric real scenes with cluttered backgrounds. Additionally, we use ResNet34 as the backbone and provide the segmentation results in Figure 2. Unlike sVORF with ViT-Base, sVORF with ResNet34 produces a coarse segmentation and still segments foreground object from complex scenes.

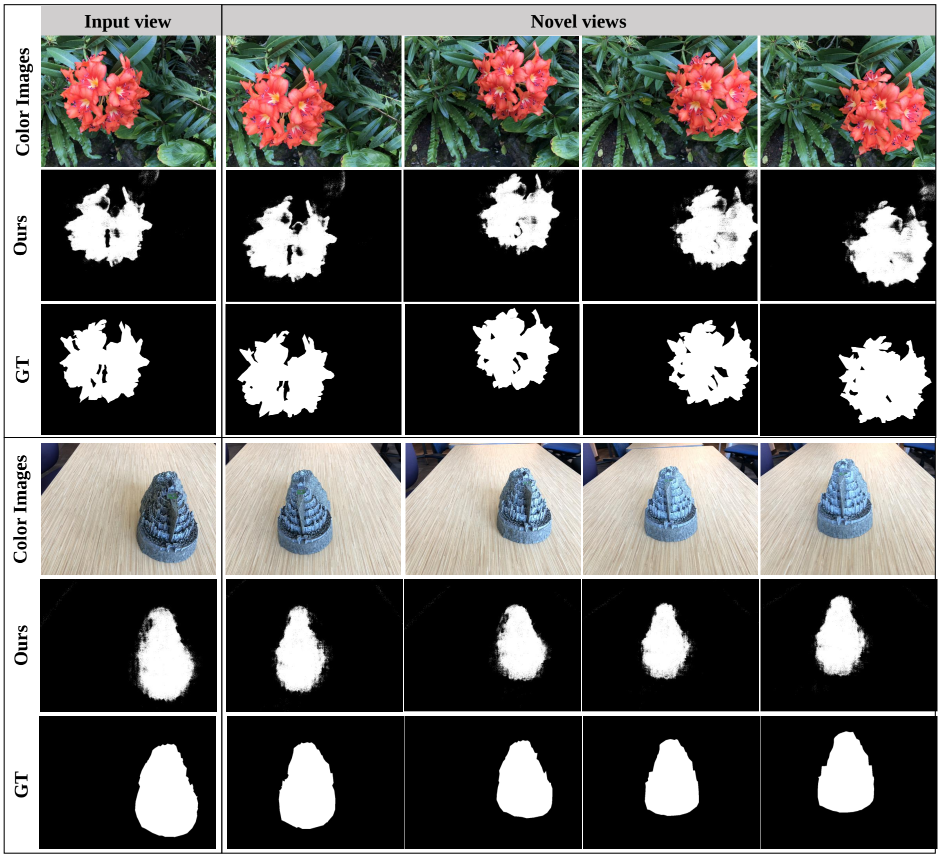

Figure 1: Qualitative results of 3D segmentation on real images.

C More Ablation Studies

This section provides more ablation studies on architecture, assessing the efficacy of the transformer-based module (Section 3.2), hypernetwork (Section 3.3) and combination module (Section 3.3), respectively. The quantitative results are reported in Table 2.

Transformer-based module We substitute the transformer with slot attention and observe that slot attention fails to achieve the decomposition task in our model, as shown in the second row of Table 2.

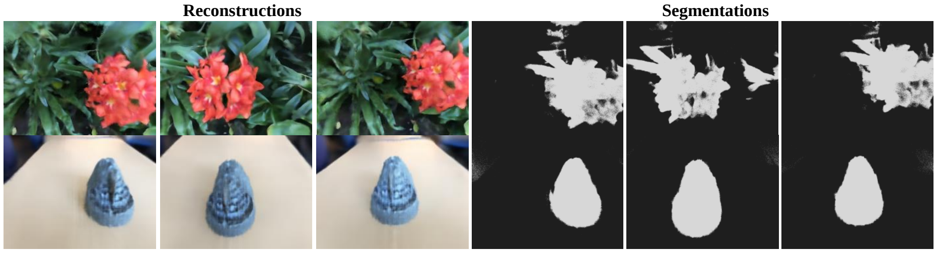

Figure 2: Qualitative results of sVORF with ResNet-34 on LLFF dataset.

Table 2: Ablation studies on the CLEVR-567 dataset.

Model	NV-ARI ↑	FG-ARI ↑
sVORF (w/o Hypernetwork)	21.6	65.9
sVORF (w Slot-Attention)	14.1	76.8
sVORF (w SM)	28.4	71.2
sVORF (ours)	**81.5**	**92.0**

Based on this comparison, we can conclude that our transformer-based module has a better scene decomposition than slot attention in our training setting.

Hypernetwork We replace the hypernetwork with directly using the slots for conditioning the radiance fields per object like uORF . As shown in the first row of Table 2, the model's performance significantly decreases. We speculate that using hypernetwork can provide stronger 3D geometric bias than directly using the slots for conditioning the radiance fields per object.

Our composition module v.s. Slot Mixers To compare our composition method with Slot Mixers decoder, we have some modifications on the SM decoder . First, we use a 3D point x instead of the target ray as the query to aggregate the weighted slot feature. Second, we transform the weighted slot feature into the corresponding radiance field. Third, based on the radiance field, we can obtain the density and color of the 3D point x. In the experiment, we find that the composition performance of the SM method is lower than our proposed composition method. Specifically, the SM method exhibits 3D-inconsistent segmentation results, as shown in the third row of Table 2. It proves that the introduction of 3D geometric bias is really important for scene decomposition.

D More Qualitative Results

Depth maps We illustrate the depth maps of sVORF on different datasets in Figure 3. The results show that our method can learn a high quality of 3D geometry.

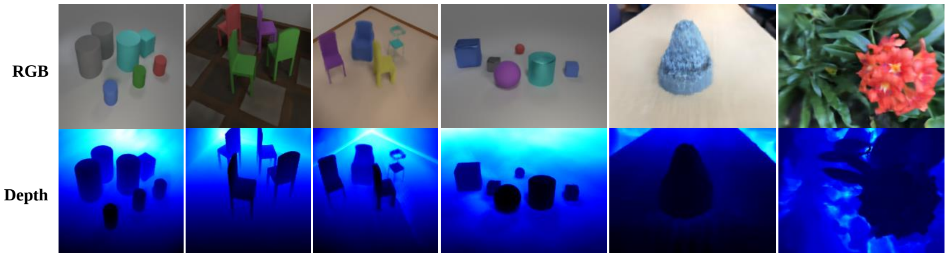

Figure 3: Reconstructed depth maps of sVORF on different datasets.

Object-level radiance fields. We provide the visualization of learned object-level radiance fields on CLEVR-567 dataset in Figure 4. It is further demonstrated that our method can achieve very clean scene decomposition.

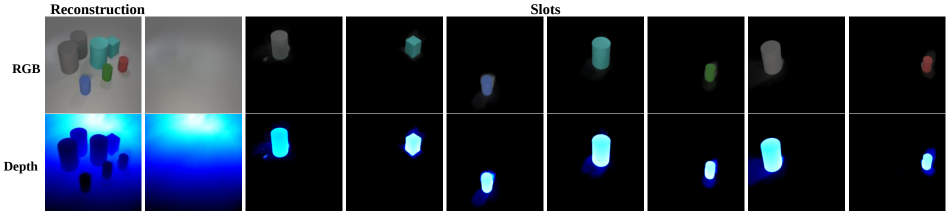

Figure 4: Visualization of learned object-level radiance fields.

Grayscale images. We provide the multi-view results of the qualitative evaluation of the model on a grayscale version of the CLEVR-567 dataset in Figure 5.

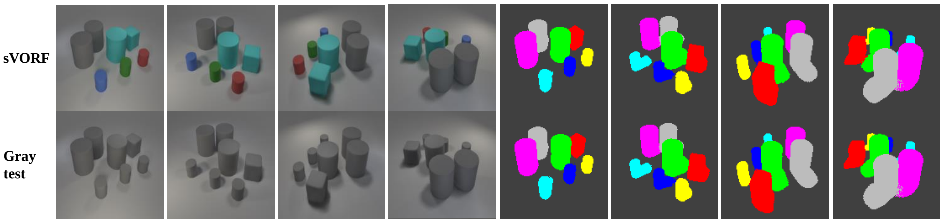

Figure 5: Multi-view qualitative results on a grayscale version of the CLEVR-567 dataset.

E Limitations

There are three limitations in sVORF. First, although sVORF can generalize to more objects at test time, as shown in Section 4.7 , the maximum slot number of test scenes is restricted by the training setting. In other words, we need know the maximum number of objects/slots in the scene, and ensure that the number of slots set is equal to or exceeds this maximum value. Second, like other 3D representation learning methods, the model's training process requires curated multi-view images, which are difficult to obtain and necessitate specialized equipment, particularly in real scenes. Finally, although our method performs well on some complex scenes, such as MSN, it is still a challenge for our method to decompose and understand real-world scenes like a human. Addressing the limitations related to the slot number, training dataset and the generalization capabilities on real scenes is a potential area for future research.