Surface-SOS: Self-Supervised Object Segmentation via Neural Surface Representation

Xiaoyun Zheng, Liwei Liao, Jianbo Jiao, *Member, IEEE*, Feng Gao, *Member, IEEE*, and Ronggang Wang, *Member, IEEE*

***Abstract*— Self-supervised Object Segmentation (SOS) aims to segment objects without any annotations. Under conditions of multi-camera inputs, the structural, textural and geometrical consistency among each view can be leveraged to achieve fine-grained object segmentation. To make better use of the above information, we propose Surface representation based Self-supervised Object Segmentation (Surface-SOS), a new framework to segment objects for each view by 3D surface representation from multi-view images of a scene. To model high-quality geometry surfaces for complex scenes, we design a novel scene representation scheme, which decomposes the scene into two complementary neural representation modules respectively with a Signed Distance Function (SDF). Moreover, Surface-SOS is able to refine single-view segmentation with multi-view unlabeled images, by introducing coarse segmentation masks as additional input. To the best of our knowledge, Surface-SOS is the first self-supervised approach that leverages neural surface representation to break the dependence on large amounts of annotated data and strong constraints. These constraints typically involve observing target objects against a static background or relying on temporal supervision in videos. Extensive experiments on standard benchmarks including LLFF, CO3D, BlendedMVS, TUM and several real-world scenes show that Surface-SOS always yields finer object masks than its NeRF-based counterparts and surpasses supervised single-view baselines remarkably. Code is available at: https://github.com/zhengxyun/Surface-SOS.**

***Index Terms*— Self-supervised learning, neural surface representation, multi-view object segmentation.**

Manuscript received 14 July 2023; revised 9 January 2024; accepted 21 February 2024. Date of publication 12 March 2024; date of current version 15 March 2024. This work was supported in part by the Outstanding Talents Training Fund in Shenzhen; in part by Shenzhen Science and Technology Program-Shenzhen Cultivation of Excellent Scientific and Technological Innovation Talents Project under Grant RCJC20200714114435057; in part by Shenzhen Science and Technology Program-Shenzhen Hong Kong Joint funding Project under Grant SGDX20211123144400001; in part by the National Natural Science Foundation of China under Grant U21B2012; and in part by Migu Cultural Technology Company Ltd., (Migu)-Peking University Meta Vision Technology Innovation Laboratory. The work of Jianbo Jiao was supported by the Royal Society under Grant IES\R3\223050 and Grant SIF\R1\231009. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Nikolaos Mitianoudis. *(Corresponding author: Ronggang Wang.)*

Xiaoyun Zheng, Liwei Liao, and Ronggang Wang are with the School of Electronic and Computer Engineering, Peking University Shenzhen Graduate School, Shenzhen 518055, China, and also with the Peng Cheng Laboratory, Shenzhen 518000, China (e-mail: xyun_z@stu.pku.edu.cn; levio@pku.edu.cn; rgwang@pkusz.edu.cn).

Jianbo Jiao is with the School of Computer Science, University of Birmingham, B15 2TT Birmingham, U.K. (e-mail: j.jiao@bham.ac.uk).

Feng Gao is with the School of Arts, Peking University, Beijing 100871, China (e-mail: gaof@pku.edu.cn).

This article has supplementary downloadable material available at https://doi.org/10.1109/TIP.2024.3374199, provided by the authors.

Digital Object Identifier 10.1109/TIP.2024.3374199

I. Introduction

GIVEN a set of multi-view images or a casually-captured video, segmentation of foreground objects is an important problem in computer vision, with downstream applications in segmentation [1], [2] and beyond, such as in video editing [3], [4], 3D scene understanding [5], [6]. Robust object segmentation can now be achieved reliably in scenarios for which large amounts of annotated data are available [7], [8], [9]. However, for less common activities, such as concerts and stage shows, it remains challenging, due to the difficulty in accessing corresponding annotated datasets. Despite some self-supervised approaches [10], [11] promising to address this problem, most of them depend on strong constraints, such as the target objects being seen against a static background, or relying on temporal supervision on video. This may result in blurry segmentation masks and false detection of segmentation boundaries.

2D images are the projections of the underlying 3D scenes. Consequently, omitting 3D information may lead to ambiguities in the task resulting from partial occlusion and background confusion, as in most single-view-based approaches. Using several cameras complicates data acquisition but only to a limited extent, as calibrations and predictable setups are available for most applications or can be computed using off-the-shelf tools such as Structure from Motion (SfM) [12]. This is also true for the emerging trend of hand-held cell phone, and static camera setups such as performance capture studios [13], [14]. These cameras are readily available, and allow for impromptu shots, as well as quick coverage of large spaces. Under conditions of multi-camera inputs, the structural, textural and geometrical consistency among each view can be leveraged to achieve fine-grained object segmentation [1]. Motivated by the above issues, we reconsider the task of self-supervised segmentation from a 3D perspective, given only 2D images of a scene from multiple viewpoints, the cross-view geometric constraints are embedded in the form of one-to-one dense mapping in 3D space, see Fig. 1 for example. This is an intrinsically challenging problem, especially when the number of views is small, or viewpoints far apart.

The emerging neural implicit representation approaches provide promising results in novel view synthesis [15], [16], [17] and high-quality 3D reconstruction from multi-view images [18], [19], [20]. The neural volume rendering approach presented in [15] and the follow-up works [21], [22] have recently shown that representing both the density and radiance

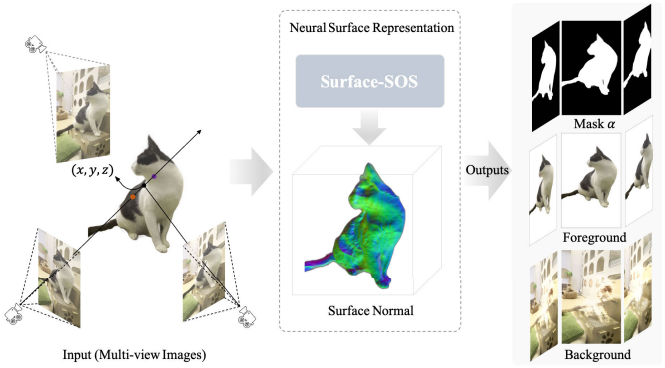

Fig. 1. We present Surface-SOS, in which multi-view geometric constraints are embedded in the form of dense one-to-one mapping in 3D surface representation. Given multi-view images as input, Surface-SOS predicts convincing results including object masks, foregrounds and backgrounds.

fields as neural networks can lead to promising novel view synthesis results from a sparse set of images. Such approaches significantly transfer multi-view information between views without explicit reconstruction of 3D geometry, neural radiance field (NeRF) [15] and its many scene-specific NeRF works will likely make a long-lasting impact on semantic scene understanding. Although this coupling indeed leads to a good generalization of novel viewing, the density is not as successful in faithfully predicting the actual geometry of the scene, often producing noisy, low-fidelity geometry approximation. These methods require precise object masks and appropriate weight initialization due to the difficulty of propagating gradients [21]. Due to this sampling imbalance, volumes that are close to the camera receive significantly more gradients [23]. This can lead to incorrect density buildup and result in floating artifacts. Moreover, it is generally difficult when the object of interest is severely occluded, with weak texture or with a similar appearance to the background. The seminal neural implicit surface representation [19], [24] substitutes the density field with a zero-level set of Signed Distance Function (SDF) in the volume rendering formulation, leading to a better approximation of the geometry while maintaining the quality of view synthesis and high-quality 3D reconstruction.

In light of this, we take one step further to investigate how to effectively segment objects from a 3D perspective. We present a framework, named Surface Representation for Self-Supervised Object Segmentation (Surface-SOS), connects object segmentation and neural surface representation to segment objects within a complex scene. We propose a neural scene decomposition scheme that decouples the 3D scene into two complementary neural representation modules for both the foreground and background: a Foreground Consistent Representation (FoCoR) module and a Background Completion (BaCo) module. By connecting the SDF-based surface representation to geometric consistency, and applying volume rendering to train this representation with robustness, we can reconstruct the foreground object geometry and background appearance from multi-view images. To accelerate the training process, and apply volume rendering to train this representation with robustness, we incorporate SDF volumetric rendering with multi-resolution hash encodings [17]. Moreover, we propose several critical training strategies for faster training convergence and better surface representation, favoring better foreground and background decomposition with fine detail.

The proposed method takes a sequence of multi-view images as input, and estimates a dense, geometrical consistent object map, as well as a textural, completed background for each view. Such a representation characterizes the compositional nature of scenes and provides additional inherent information, thus benefiting 3D scene decomposition. We validate the effectiveness of Surface-SOS using multi-view datasets and monocular stereo video, including public benchmarks for real-world forward-facing datasets (LLFF [25]), object-centric datasets BlendedMVS [26], Common Objects in 3D (CO3D [27]), and TUM dataset [28]; and real-world RGB video captures from [29]. Extensive experiments show that Surface-SOS performs better than the state-of-the-art (SOTA) image-based object (co-) segmentation frameworks and NeRF-based object segmentation methods. Even without auxiliary inputs such as object mask, Surface-SOS can effectively recover dense 3D surface structures from multi-view images that lead to photo-realistic rendering results and high-quality segmentation maps.

In summary, this paper makes the following contributions:

- We present Surface-SOS, a new self-supervised approach that leverages neural surface representation to break the dependence on large amounts of annotated data and strong constraints to achieve superior performance.
- We design a 3D scene decomposition scheme containing two complementary neural representation modules for both foreground and background. By leveraging the multi-resolution hash grid SDF, Surface-SOS can generate compact geometric surfaces, and produces finer object segmentation compared to its NeRF-based counterparts.
- We introduce several critical strategies for faster convergence and better surface representations. Our proposed framework can be implemented on general benchmarks for forward-facing, object-centric, indoor, and real-world dynamic scenarios.
- Extensive experiments and ablation studies justify the design of each component and demonstrate that Surface-SOS yields finer object segmentation than its NeRF-based counterparts. It also remarkably improves single-view methods by simply adding original masks as an additional input and generating fine-grained segmentation.

II. Related Work

A. Single View Object Segmentation

Object segmentation is a longstanding problem in computer vision. Most object segmentation algorithms are fully supervised [7], [31] and require large annotated datasets containing pairs of images and labels [32]. Our goal is to train a purely self-supervised method without either segmentation or object bounding box annotations. Many approaches take advantage of the motion patterns of objects as complementary cues [33], [34], which use a two-stream network to process the RGB image and the corresponding optical flow separately and fuse the results in the end. To avoid the expensive computation of optical flow, some work [35], [36] utilizes higher-order

spatial and temporal relations between video frames to bring more comprehensive content understanding. However, these motion-based segmentation methods are prone to accumulate errors calling for a new system with high accuracy performance and robustness on each frame. Furthermore, motion-based object segmentation relies on masks or uses task-specific training datasets, which lack the capability of preserving fine details, e.g., human hairs and animal fur.

Image matting deals with the problem of estimating an RGBA foreground (color image + alpha matte) and a background color image from a given image [37], [38], [39]. Mathematically, an image I can be viewed as the linear combination of a foreground F and a background B through an α coefficient: $I = \alpha F + (1-\alpha)B$. With the help of trimaps, image matting predicts a detailed alpha matte which can be used to recover the mixing factor of foreground and background [39], [40]. Extensive research has provided promising performance on deep learning-based video matting [9], [41]. However, existing supervised deep models require enormous manual annotations for training. Furthermore, if the training data do not adequately cover the sampling variation, the trained model may be biased and may not generalize well for images that do not correlate strongly with the training data. It still in most cases generated and augmented the training samples by compositing the images with various background images and foreground, while it makes the synthesized images into unreal scenarios, and produces unacceptable alpha matte.

The recent video layered representations leverage the power of deep neural networks to separate a moving object in a video from its background by representing a video as a composition of layers with simpler motions [10], [11]. They use neural rendering and fit layer models to images by optimizing transformations to minimize reconstruction loss. However, with a large number of layer decompositions that could completely reconstruct the video while outputting nonsensical group separations, this is a difficult problem to solve. The Layered Neural Atlases (LNA) [42] adopts multi-layer perceptrons (MLPs) to decompose and map the video into sets of 2D atlases. Consequently, MLP-based atlases lead to better decomposition than a standard fixed pixel grid atlas, owing to the image representations being continuous with respect to spatial or spatio-temporal pixel coordinates. As mentioned above, the ideas of layer decomposition have demonstrated the effectiveness of motion as input or supervision for segmentation. However, motion signals can be uninformative or even dishonest in cases such as deformable objects and objects with reflections or occlusions, resulting in unacceptable segmentation.

B. Co-Segmentation Approaches

Co-segmentation is the task of detecting and segmenting the common objects from an image pair and by extension to more images [43]. The key assumptions of these methods are the observation of a common foreground region, or objects with the same appearance characteristics, as opposed to a background with higher variability across images. Initially, a deep dense conditional random field [44] is applied to the co-segmentation task. They use a co-occurrence map to measure the objectness for object proposals, and the similarity evidence for proposals is generated by a selective search which uses scale-invariant feature transform (SIFT) feature. An end-to-end training framework [45] introduces convolutional neural network (CNN) to jointly detect and segment the common object from a pair of images. A later attention-based method [46] takes the encoded features to pay attention to the common objects via a semantic attention learner. Most recently, a new co-segmentation framework [47] based on the deep features extracted from a pre-trained Vision Transformer (ViT) has been proposed and achieves better results on object co-segmentation and part co-segmentation.

C. Multi-View Object Segmentation

The above approaches to object segmentation usually require the user to provide information about background or foreground in advance. To avoid ambiguity between foreground and background model, many researchers have attempted to solve this problem automatically by using additional information such as stereo cues. Early attempts [48], [49] to segment an object from multiple views by combining photometric information with depth information from stereo images. These approaches provide much better results than methods relying solely on color information, but they are designed for stereo imaging systems and cannot be readily applied to systems with more than two cameras. Some approaches for multi-view segmentation transfer information between views without explicit representations of 3D geometry [1], [50]. The first attempt to solve the multi-view segmentation problem uses a silhouette-based algorithm [50]. A later method [51] focuses on probabilistic occupancy along viewing lines. An inter-view consistent approach links superpixels between images [1], where geometric cues are propagated using camera parameters to ensure consistency between views. This approach adopts the assumption that color values of a foreground object are different from those of the background region. Indeed, it then becomes difficult to rely on shared appearance models of the object between views while parts of the background seen from several viewpoints will present similar aspects. In contrast, our approach aims towards delicate segmentation within a complex scene, by combining the neural implicit representation power from multi-view images in an end-to-end and self-supervised manner.

D. Neural Implicit Representation

The neural implicit representation has shown its effectiveness in novel view synthesis [15], [16], [17] and high-quality 3D reconstruction from multi-view images [18], [19], [20]. The neural implicit exploit the coordinate-based representation to model the scene by querying points attributes with their 5D coordinates (x, y, z, θ, ϕ). Such representation significantly improves traditional image-based modeling or rendering in an end-to-end and self-supervised manner. NeRF [15] and its many scene-specific NeRF works will likely make a long-lasting impact on semantic scene understanding. In particular, Semantic-NeRF [21] adds an extra head to NeRF

to predict semantic labels at any 3D position. As a supervised approach, Semantic-NeRF requires semantic labels for full supervision. The recent work NeRF-SOS [52] presents a framework for learning object segmentation in complex real-world scenes by using a collaborative contrastive loss at the appearance segmentation and geometry segmentation levels. However, the optimization process for training is affected by the conflicting update directions, contrastive loss, and ViT semantic feature extraction, which remains a notorious challenge in multi-task learning. RFP [53] is an unsupervised multi-view image segmentation using radiance field propagation with a bidirectional photometric loss to guide the reconstruction of semantic field. This approach adopts the assumption that there is no object motion in the scene, or color values of a foreground object are different from those of the background region. However, given only a set of calibrated input images, the NeRF reconstruction problem is ill-posed. NeRF acquisition typically suffers from background collapse, creating near-camera floating artifacts on the edges of the captured scene. Several works [23], [54] observed and proposed solutions to the problem of floaters and background collapse. The main insight is that the floating artifacts occur due to a higher density of samples in regions near cameras [23]. Similarly, without visual cues, the model has lost the perception of the object. Nevertheless, extending the semantics and discovery of object decompositions with NeRF is not trivial, as they cannot extract high-quality surfaces since the geometry representation does not contain surface constraints. The techniques are generally difficult when the object of interest is highly occluded, has a weak texture, or has a similar appearance to the background.

Early work focused on predicting the geometry of shapes using occupancy fields [55] or SDF [56]. NeuS [19] represents the 3D surface as an SDF for high-quality geometry reconstruction. This allows for differentiable rendering by tracing rays through the scene and integrating over them. However, the explicit integration also makes this approach very computationally intensive, training NeuS is very slow, and it only works for static scene reconstruction. To overcome the slow training time of deep coordinate-based MLPs, Instant-NGP [17] proposes a multi-resolution hash encoding and proves its effectiveness to speed up. With this in mind, we present a self-supervised learning framework for multi-view object segmentation by leveraging the geometric consistency of instant neural surface representation. Unlike previous works that assume a static camera or background, we allow the representation network to deform space along with the parallax or pose changes, then reconstruct the geometry and appearance of the foreground, as well as a textured, complete background.

III. Proposed Method

A. Overview

Surface-SOS takes a sequence of multi-view images as input, and estimates a dense, geometrical consistent object segmentation map, as well as a textural, completed background for each view. Fig. 2 shows an overview of our method. To tackle this challenging task by leveraging the existence of geometric consistency of the one-to-one dense mapping in 3D space, we decouple the scene into two complementary neural scene representation modules: a Foreground Consistent Representation (*FoCoR*) module and a Background Completion (*BaCo*) module. We build our scene representation modules upon the SDF-based neural surface representation, and incorporate multi-resolution hash encodings for training acceleration. We introduce geometric and photometric losses to train the network in a self-supervised manner. Moreover, we propose several critical training strategies for faster training convergence and better surface representation, which favors better foreground and background decomposition with fine detail. Given a 3D spatial point, we concatenate its queried feature from the multi-resolution hash grid and its 3D position as input to the SDF-based networks. The outputs of SDF network are combined with the viewing direction and further fed into the RGB network. For FoCoR, the RGB network predicts color and alpha values for the foreground, while for BaCo, the network predicts the background color.

B. Preliminary

NeuS. Given a set of posed multi-view images, the scene of the object is represented by two functions: a signed distance field $f(\mathbf{p}) : \mathbb{R}^3 \rightarrow \mathbb{R}$ that maps a spatial position $\mathbf{p} \in \mathbb{R}^3$ to its signed distance to the object, and a radiance field $c(\mathbf{p}, \mathbf{v}) : \mathbb{R}^3 \times \mathbb{S}^2 \rightarrow \mathbb{R}^3$ that encodes the color associated with a point $\mathbf{p} \in \mathbb{R}^3$ and a view direction $\mathbf{v} \in \mathbb{S}^2$. The surface $\mathcal{S}$ of the object can be obtained by extracting the zero-level set of the SDF $\mathcal{S} = \left\{\mathbf{p} \in \mathbb{R}^3 \mid f(\mathbf{p}) = 0\right\}$. Then the object is rendered into an image by volume rendering. Specifically, for each pixel of an image, we sample n points $\{p(t_i) = \mathbf{o} + t_i v | i = 0, 1, ..., n-1\}$ along its camera ray, where $\mathbf{o}$ is the center of the camera and $\mathbf{v}$ is the view direction. By accumulating the SDF-based densities and colors of the sample points, we can compute the color $\hat{C}$ of the ray. As the rendering process is differentiable, NeuS can learn the signed distance field $f(\mathbf{p})$ and the radiance field $c(\mathbf{p}, \mathbf{v})$ from the multi-view images. However, the training process of NeuS is very slow, taking about 8 hours on a single GPU.

Multi-resolution Hash Encoding. To overcome the slow training issue of deep coordinate-based MLPs (which is also a major issue for slow performance of NeuS), multi-resolution hash encoding was proposed [17] and shown to be effective. Specifically, it assumes that the object to be reconstructed is bounded in multi-resolution voxel grids. The voxel grids at each resolution are mapped to a hash table with a fixed-size array of learnable feature vectors. For a 3D position $\mathbf{p} \in \mathbb{R}^3$, it obtains a hash encoding at each level $h^i(p) \in \mathbb{R}^d$ (d is the dimension of a feature vector, $i = 1, \ldots, L$) by interpolating the feature vectors assigned at the surrounding voxel grids. The hash encodings at all L levels are concatenated into $h(p) = \{h^i(p)\}_{i=1}^{L} \in \mathbb{R}^{L \times d}$ to be the multi-resolution hash encoding.

C. Neural Scene Decomposition via Hash-Encoded SDF

FoCoR Module. For every image, given a 3D position $\mathbf{p}(x, y, z) \in \mathbb{R}^3$ in the rough foreground object, we map its

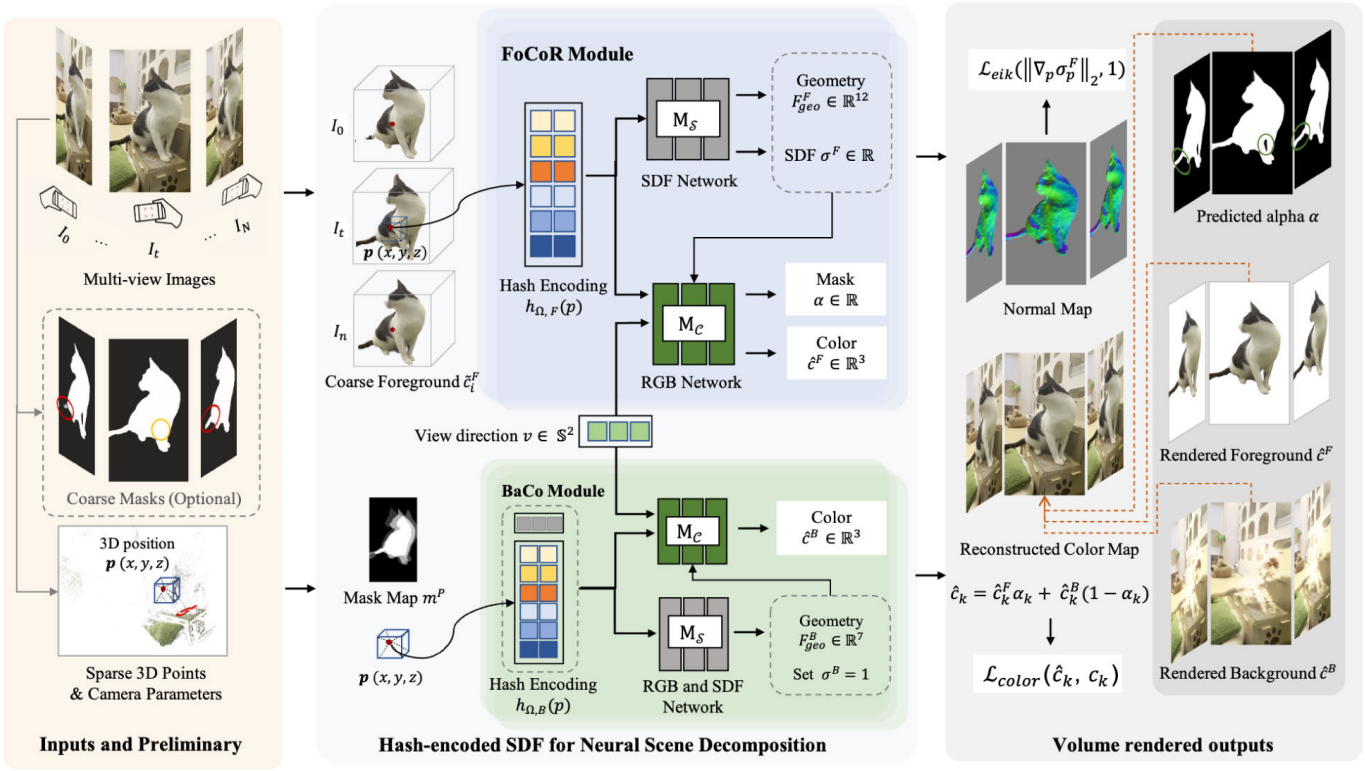

Fig. 2. **Method overview**. For the scene captured by N images $\{I_i\}_{i=1}^{N}$, we use COLMAP [30] and Mask-RCNN [7] to get sparse 3D points and coarse object masks as co-inputs, and predict a dense, geometrical consistent object map, as well as a textural, completed background for each image. Note that the coarse mask is optional and merely expedites the convergence of 3D surface representation. Moreover, by introducing coarse masks as additional input, Surface-SOS is able to refine segmentation remarkably (see the under-segmentation and over-segmentation highlighted in red and yellow, respectively). Surface-SOS consists of two complementary representation modules: a Foreground Consistent Representation (FoCoR) module and a Background Completion (BaCo) module. **FoCoR**: For every image, given a 3D point $p(x, y, z)$, we concatenate its queried feature from the multi-resolution hash grid as the input to the SDF network. The SDF network outputs the geometry feature and SDF value, which are combined with the viewing direction and further fed into the RGB network to generate RGB value for the foreground, as well as the alpha α prediction. **BaCo**: Given a sequence of multi-view images, we concatenate its static features from the multi-resolution hash grid and its 3D position $p(x, y, z)$ as the input to the SDF network. Here, we crop the foreground from the probability map region m^P by setting the SDF value to a positive number (e.g. 1.0). Then the SDF value σ^B and geometry feature vectors $\mathbf{F}_{geo}^B$ are combined with the viewing direction $\mathbf{v} \in \mathbb{S}^2$ and further fed into the RGB network $\mathrm{M}_\mathcal{C}$ to generate the RGB value for the background c^B. After removing the foreground from the probability map m^P, even though some parts of the background were occluded in the original view, the other views of the scene provide sufficient textural/structural information to complete the missing background. All parts of the proposed pipeline are trained end-to-end with the geometric and photometric losses in a self-supervised manner with the original input images.

multi-resolution hash encodings $h_\omega(p) \in \mathbb{R}^{L\times d}$ with learnable hash table entries ω. We concatenate its acquired feature vectors from the multi-resolution hash grid and the 3D position as the input to the SDF network $\mathrm{M}_\mathcal{S}$, which consists of a shallow MLP:

$$(\sigma^F, \mathbf{F}_{geo}^F) = \mathrm{M}_\mathcal{S}\left(\mathbf{p}, h_{\Omega,F}(\mathbf{p})\right). \tag{1}$$

The SDF network outputs the SDF value $\sigma^F \in \mathbb{R}$ and geometry feature vectors $\mathbf{F}_{geo}^F \in \mathbb{R}^{12}$, which are combined with the viewing direction $\mathbf{v} \in \mathbb{S}^2$ and further fed into RGB network $\mathrm{M}_\mathcal{C}$ to generate RGB value for the foreground object.

The normal $\mathbf{n}$ of the point $\mathbf{p}$ can be computed as $\mathbf{n} = \nabla_\mathbf{p}\sigma^F$ by the gradient of the SDF. We observe that the RGB network is biased to output similar colors for neighboring sample points when their corresponding surface normal is close. Adding normals to the input encourages the reconstructed surface to be smoother, especially for texture-less areas. Eventually, we feed the normal $\mathbf{n}$ with the SDF value σ^F, the geometry feature $\mathbf{F}_{geo}^F$, the point $\mathbf{p}$, and the ray direction $\mathbf{v}$ to the RGB network $\mathrm{M}_\mathcal{C}$, then the foreground appearance and alpha prediction is formulated as

$$c^F = \mathrm{M}_\mathcal{C}(\mathbf{p}, \mathbf{n}, \mathbf{v}, \sigma^F, \mathbf{F}_{geo}^F). \tag{2}$$

$$\alpha = \mathrm{M}_\mathcal{C}(\mathbf{p}, \mathbf{n}, \mathbf{v}, \sigma^F, \mathbf{F}_{geo}^F). \tag{3}$$

BaCo Module. Training neural surface representation network without a background model can lead to floaters (uncontrolled object surfaces) in free space [19]. One of the main reasons for this problem is the floaters in the background color, which do not affect the rendering quality and therefore cannot be optimized when training with only photometric loss. Since we want to segment and decouple the foreground and background components of the scene, we circumvent this problem by applying a static background module $c(\mathbf{p}, \mathbf{v}) : \mathbb{R}^3 \times \mathbb{S}^2 \rightarrow \mathbb{R}^3$ to inpaint static background from the cropped mask region M_p. The mask $m^P \in [0, 1]$ indicates the likelihood that a point belongs to the swept volume of the foreground in the scene. The SDF network outputs the SDF value $\sigma^B \in \mathbb{R}$ and geometry feature vector $\mathbf{F}_{geo}^B \in \mathbb{R}^7$. Here, we crop the foreground from the probability map region m^P by simply setting the SDF value to a positive number

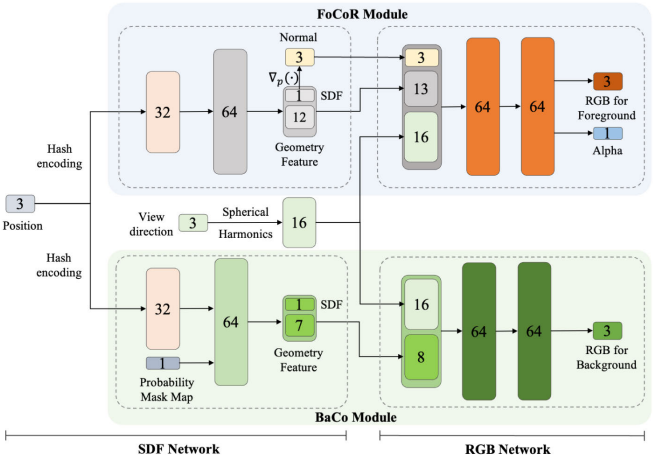

Fig. 3. A visualization of the architecture of FoCoR and BaCo module.

(e.g. 1.0). Since we assume a unimodal (i.e. bell-shaped) density distribution centered at 0 [19], an area with a large SDF value is less likely to be sampled, and therefore appears transparent. Then the density value σ^B and the geometry feature vector $\mathbf{F}^B_{geo}$ are combined with the viewing direction $\mathbf{v} \in \mathbb{S}^2$, to be further fed into the RGB network $\mathrm{M}_\mathcal{C}$ to generate RGB value for the background c^B. After removing the foreground from the probability map m^P, even with parts of the background occluded in the original view, the other views of the scene provide sufficient textural/structural information to complete the missing background.

$$c^B = \mathrm{M}_\mathcal{C}(\mathbf{p}, \mathbf{n}, \mathbf{v}, m^P, \sigma^F, \mathbf{B}^B_{geo}). \tag{4}$$

Finally, the RGB color of a 3D position $\mathbf{p}$ can be reconstructed by alpha-blending the corresponding points:

$$c = \alpha c^F + (1-\alpha)c^B. \tag{5}$$

As shown in Fig. 3, the network architecture consists of the following components: (a) two multi-resolution hash grids with 16 levels of different resolutions ranging from 16 to 2048; (b) two SDF networks modeled by a 1-layer MLP with 64 hidden units; (c) an RGB network modeled by a 2-layer MLP with 64 hidden units for consistent foreground object; (d) an RGB network modeled by a 2-layer MLP with 64 hidden units for static background.

D. Supervision and Volume Rendering Outputs

1) Volume Rendering: To learn the parameters of the neural SDF and color field, we apply the unbiased volume rendering scheme to render images from the Hash-encoded SDF representation. Given a pixel, we sample n points $\{p(t_i) = \mathbf{o} + t_i v | i = 0, 1, \ldots, n\}$ along its camera ray, where $\mathbf{o}$ is the camera center and $\mathbf{v}$ is the unit direction vector of the ray.

To obtain discrete counterparts of the opacity and weight function, we still need to adopt an approximation scheme, which is similar to the composite trapezoid quadrature. By using opaque density σ^F, the alpha values α are defined in discrete form by

$$\alpha_i = \max\left(1 - \frac{\Phi_b\left(f\left(\mathrm{p}\left(t_{i+1}\right)\right)\right)}{\Phi_b\left(f\left(\mathrm{p}\left(t_i\right)\right)\right)}, 0\right), \tag{6}$$

where $\Phi_b(x) = 1/(1+e^{bx})$ known as the cumulative density distribution, b is a trainable hyperparameter and gradually increases to a large number as the network training converges.

By accumulating the redefined α values and colors of the sample points, the final color $\hat{c}$ along the ray is computed via the approximation scheme as

$$\hat{c}_(\mathbf{o}, \mathbf{v}) = \sum_{i=1}^{n} T(t_i)\alpha(t_i)c(\mathbf{p}(t_i), \mathbf{v}), \tag{7}$$

where $T(t_i)$ is the discrete accumulated transmittance defined by $T\left(t_i\right) = \prod_{j=0}^{i-1}\left(1-\alpha\left(t_j\right)\right)$, $c(p(t), \mathbf{v})$ means the color at the point $\mathbf{p}$ along the viewing direction $\mathbf{v}$.

Additionally, we adopt a ray marching acceleration strategy [17] to maintain an occupancy grid that roughly marks each voxel grid as empty or non-empty. The occupancy grid can effectively guide the marching process by preventing sampling in empty spaces, and accelerate the volume rendering process.

2) Training and Supervision: With the multi-view images as the main supervision signal used to train our system, we introduce photometric and geometric loss to supervise their training. Specifically, we optimize our neural networks and inverse standard deviation by randomly sampling a batch of pixels and their corresponding rays in the world space $P = \{c_k, M_k, \mathbf{o}_k, \mathbf{v}_k\}$, $k \in \{1, \ldots, m\}$ from an image in every iteration, where m denotes the batch size, c_k is its pixel color and $M_k \in \{0, 1\}$ is its coarse mask value. The final joint loss function is defined as

$$\mathcal{L} = \mathcal{L}_{\text{color}} + \lambda_e \mathcal{L}_{\text{Eikonal}} + \lambda_s \mathcal{L}_{\text{sparsity}}. \tag{8}$$

First, we minimize the distance between rendered pixels $\hat{c}_k$ and the ground truth pixels c_k, by using a color loss $\mathcal{L}_{\text{color}}$ defined as

$$\mathcal{L}_{\text{color}} = \frac{1}{m}\sum_k \mathcal{R}\left(\hat{c}_k, c_k\right). \tag{9}$$

Here we choose the L2 loss as $\mathcal{R}$, which is shown to be robust to outliers and stable in training.

An important property of SDF is its unit norm. The Eikonal term [57] is therefore added to regularize the learned signed distance field. Specifically, Eikonal loss $\mathcal{L}_{\text{Eikonal}}$ is added on the normal of sampled points:

$$\mathcal{L}_{\text{Eikonal}} = \frac{1}{mn}\sum_{k,i}\left(\left\|\mathbf{n}_{k,i}\right\| - 1\right)^2, \tag{10}$$

where i indexes the i th sample along the ray with $i\{1, \ldots, n\}$, and n is the number of sampled points. $\mathbf{n}_{k,i}$ is the normal of a sampled point.

Additionally, the property of accumulated transmittance in volume rendering means that the invisible query samples behind visible surfaces lack supervision. To address this issue of floaters and background collapse, a sparsity regularization

term [58] is incorporated to generate compact geometric surfaces.

$$\mathcal{L}_{\text{sparsity}} = \frac{1}{mn} \sum_{k,i} exp\left(-\tau \cdot |\sigma^F|\right)^2, \tag{11}$$

where $|\sigma^F|$ is the absolute SDF value of sampled point, τ is a hyperparameter to re-scale the SDF value. This term will encourage the SDF values of the points behind the visible surfaces to be far from 0. When extracting 0-level set SDF to generate mesh, this term can avoid uncontrollable free surfaces. The sparsity loss $\mathcal{L}_{sparsity}$ prevents duplicate representations in the foreground, and further encourages the many-to-one mapping of scene points to the sample points.

We additionally introduce an optional mask loss $\mathcal{L}_{\text{mask}}$ during the initial train phase. We encourage the predicted alpha values to match a coarse input mask that identifies which regions should decouple the foreground and background scenes. We note that we do not require input masks to be precise, as they are used only for initializing the alpha mapping. And the majority of training happens without this loss, allowing the network to correct for errors. The alpha loss $\mathcal{L}_{\text{mask}}$ is defined as:

$$\mathcal{L}_{\text{mask}} = BCE(M_k, \alpha_k), \tag{12}$$

where BCE is the binary cross entropy loss.

IV. Experiments

A. Experimental Settings and Implementation Details

1) Datasets and Evaluation Metrics: We validate the effectiveness of the proposed Surface-SOS on both multi-view benchmark datasets and monocular stereo video data. We provide qualitative and quantitative comparisons with the SOTA object segmentation methods on four publicly available datasets, and choose some representative scenes: 1) LLFF datasets [25] contain three forward-facing scenes {*Flower, Fortress, horns*} with 30 to 62 roughly forward-facing images; 2) BlendedMVS datasets [26] contain two object-centric scenes {*5a6, 5c3*} with 27 to 110 images; 3) CO3D datasets [27] contain two common objects {*Bicycle, Backpack*} captured with 100 to 201 images; and 4) two scenes {*Teddy bear, Plant*} in the 3D object reconstruction category of the TUM datasets [28], sampling with sequences ranging from 175 to 259 frames. We manually labeled all views as faithful binary mask annotations to provide a quantitative comparison for all methods and used them to train Semantic-NeRF [21]. However, these public datasets are either designed for novel view synthesis [25], specific domains [26], or videos of static scenes [27], [28]. To validate the effectiveness of our approach, we further evaluate on additional more challenging datasets. Specifically, we capture custom stereo video datasets for evaluation, including hand-held phones, and static camera setups such as performance capture studios. Our dataset consists of both static *Dance* and dynamic scenes *Cat* with a gentle amount of object motion. We also leverage three common scenes {*Kevin, Texting, Boy*} captured by video sequences from [29]. We sample the videos and obtain sequences ranging from 73 to 180 frames.

As for quantitative evaluation, we use the Sum of Absolution Difference (SAD), Mean Square Error (MSE), mean pixel accuracy (Acc.), and mean Intersection over Union (mIoU) as our metrics. Acc. measures the proportion of pixels that have been assigned to the correct region, and mIoU is the ratio between the area of the intersection between the ground-truth segmentation mask and the prediction.

2) Comparison Methods: Given multi-view images or a casually-captured video, we target to output the corresponding alpha foreground as a segmentation mask. Therefore, several object segmentation baselines are adopted for comparisons: 1) single-view supervised segmentation SAM [32]; 2) image-based object co-segmentation method DINO-CoSeg [47]; and 3) NeRF-based methods, including NeRF-SOS [52], RFP [53] and supervised Semantic-NeRF [21] trained with annotated masks. On the other hand, video-based foreground matting RVM [9], and the untrained network-based methods LNA [42] for video layers decomposition are included for a more comprehensive comparison of video sequences.

3) Implementation Details: Given a sequence of multi-view images, we first perform an SfM [12] reconstruction using an open-source software COLMAP [30] to estimate the camera poses and sparse 3D points of the scene. This step provides us with intrinsic and extrinsic camera parameters as well as a sparse point cloud reconstruction. To expedite the convergence of 3D object surface representation, we apply Mask R-CNN [7] to segment out the most common foreground in each view independently. SAM [32] is a general segmentation model trained on a diverse, high-quality dataset of over 1 billion masks, it can produce high-quality object masks from input prompts such as points or object bounding boxes. DINO-CoSeg [47] is an image-based object co-segmentation method as it takes a pair of images as input and automatically co-segment semantically common foreground objects. Semantic-NeRF [21] is a supervised NeRF-based approach as it takes annotated labels as input to supervise a semantic branch for object separation. Thus we feed the ground truth labels as input to these methods where the official implementations are used. NeRF-SOS [52] is a self-supervised framework, in which the collaborative contrastive loss is implemented upon the original NeRF [15], and segmentation results are based on K-means clustering. RFP [53] is one of the first real-scene NeRF-based approaches for unsupervised multi-view image segmentation. We share the same task of scene object segmentation in 3D perspective with multi-view settings. RFP [53] relies on unsupervised single image segmentation algorithms to get good initialization, thus we provided the initial masks of IEM [59] as input which was used in RFP [53]. Here we only present the results of RFP on the LLFF dataset, as the official implementation for the other datasets is still not available. Our whole system and all the experiments are implemented on a machine with a single NVIDIA GeForce RTX3090 GPU. We train our models using the ADAM optimizer with a learning rate of 0.01. For each scene representation, we train our model for 40k iterations, which takes around 25 minutes.

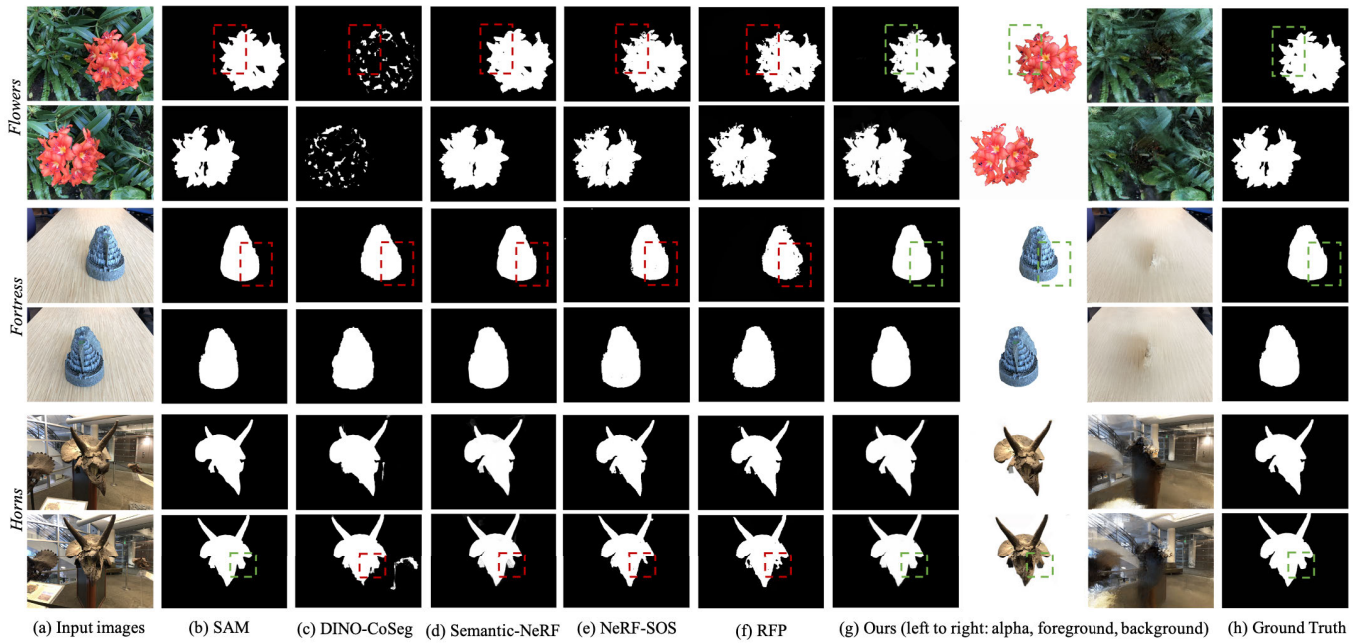

Fig. 4. Comparison on the forward-facing scenes *Flower, Fortress*, and *horns* from LLFF dataset [25]. In the third column, DINO-CoSeg [47] mistakenly matches several discrete patches, as DINO has higher activation on just a few tokens, which may lead to view-inconsistent and disconnected co-segmentation results. Compared to SAM [32] and DINO-CoSeg, our results have more accurate edges, since our network can exploit multi-scale geometry features to better capture the matte objects. Compared with NeRF-based methods (i.e. Semantic-NeRF [21], NeRF-SOS [52], and RFP [53]), Surface-SOS (g) produces view-consistent masks with finer details and no holes in the interior of objects.

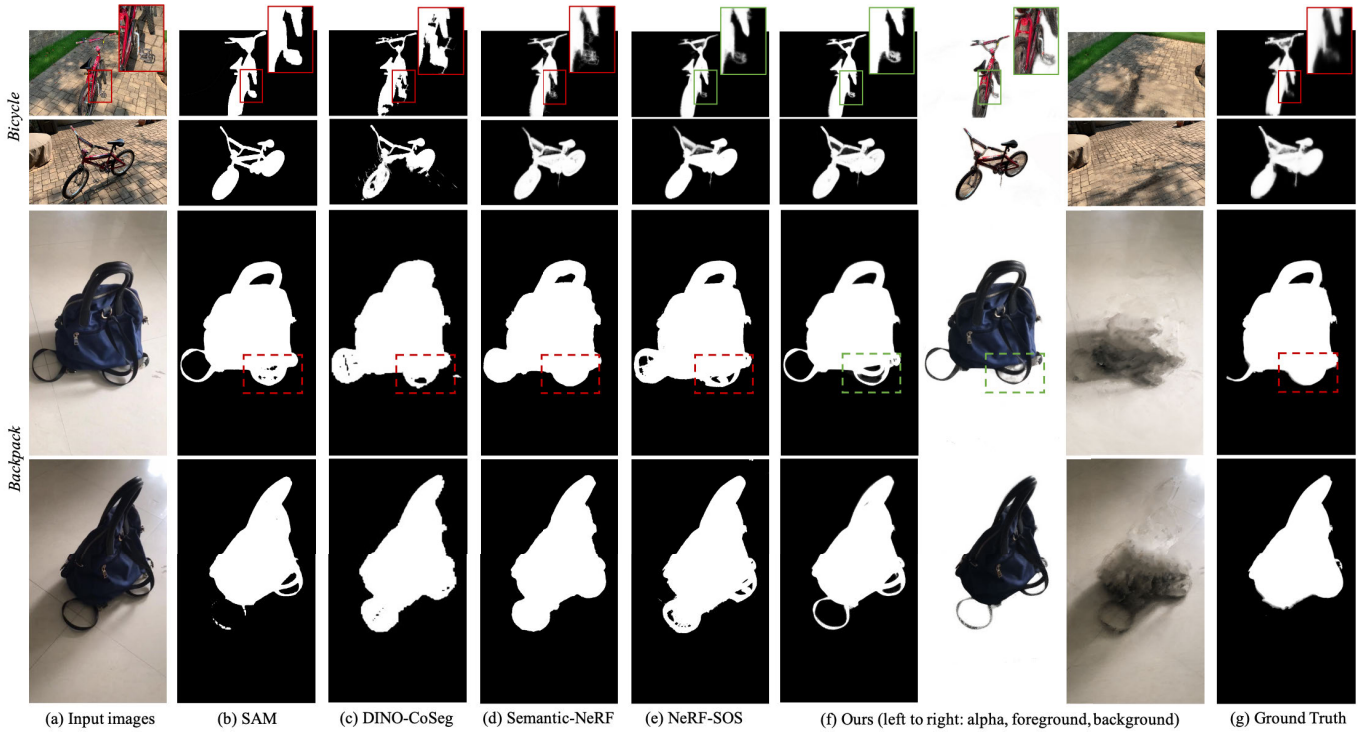

Fig. 5. Qualitative comparisons on object-centric scenes *Biclcle* and *Backpack* from CO3D data [27]. Despite SAM [32] providing fine-grained boundary information it is noisy and misses more valid detection than ours. Whereas the proposed method achieves high-quality geometric and textural consistent foreground maps without inducing noise, e.g., it can recover the complex structures of the bicycle frame and render detailed textures in the *Bicycle* example.

B. Qualitative Results

1) Comparisons With SOTA on Static Scenes: For static scene object segmentation, we present examples of the comparison with different baselines on a variety of scenes. CO3D provides ground-truth label maps using PointRend [31], we create binary masks annotation of other datasets for evaluations and the Semantic-NeRF training with publicly available annotation tool labelme.[1]

[1]https://github.com/wkentaro/labelme

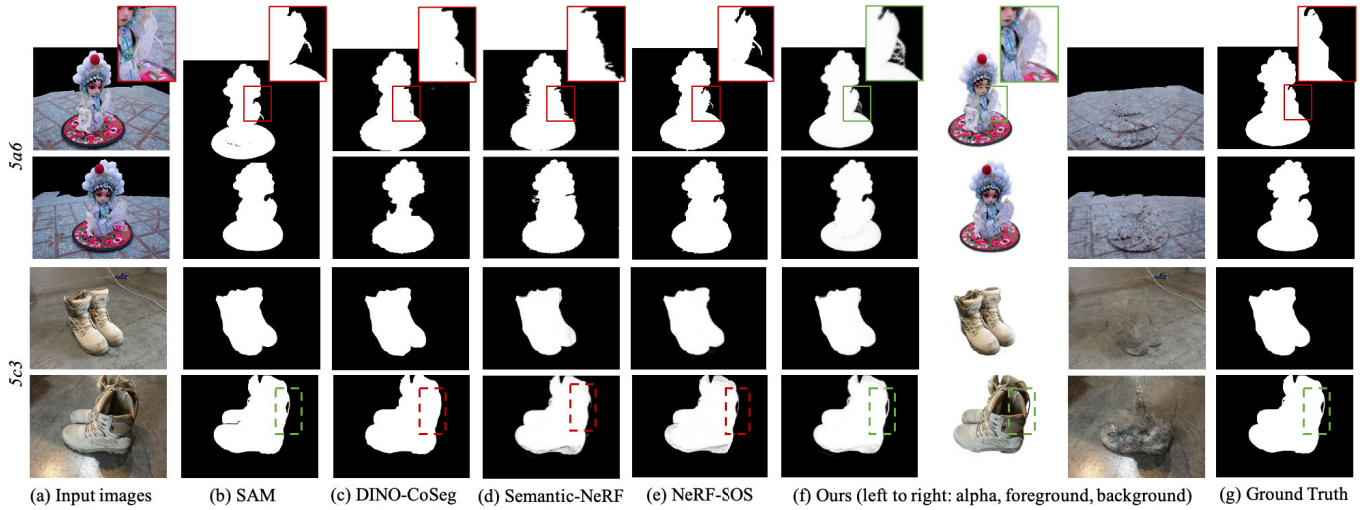

Fig. 6. Qualitative comparison on the object-centric scenes *5a6* and *5c3* from BlendedMVS dataset [26]. Surface-SOS produces more view-consistent masks than other NeRF-based methods. It even generates finer details than the supervised Semantic-NeRF [21] and SAM [32] (see the openwork sleeve in the top row and the shoelace in the bottom row).

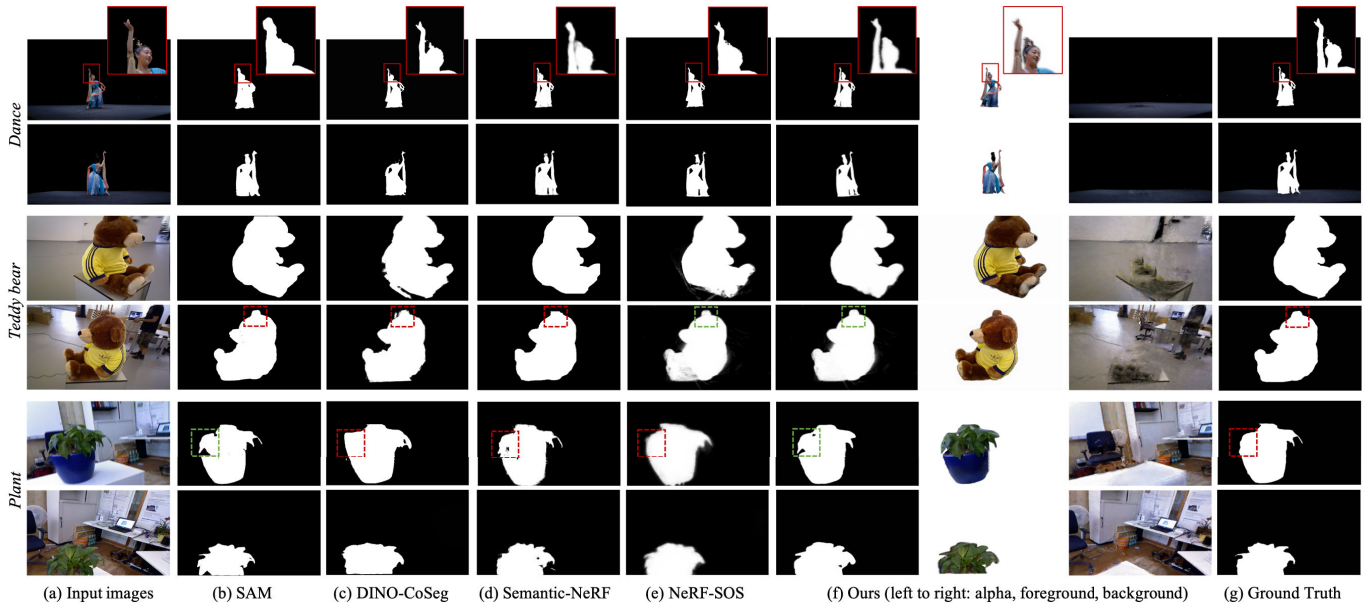

Fig. 7. Qualitative comparison on the object-centric scenes *Dance, Teddy bear*, and *Plant*. These examples span a wide range of human and non-human species on the complex scene, suggesting the superiority of our proposed methods in the generation of geometrical consistent foreground matte, as well as a textural, completed background.

LLFF [25] Scenes. As seen in Fig. 4, compared to single-view supervised segmentation SAM [32] and image-based object co-segmentation DINO-CoSeg [47], our method presents more accurate edges, due to the exploitation of multi-scale geometry features to better capture the matte objects. Compared to NeRF-based methods, including NeRF-SOS [52], RFP [53] and supervised Semantic-NeRF [21], Surface-SOS produces a more complete foreground matte with no holes in the interior of objects, due to the high-quality geometry representation with the surface constraints.

Object-centric Scenes. Here we use CO3D [27], BlendedMVS [26], TUM [28] datasets, and the *Dance* scene captured by static camera setups.

As shown in Fig. 5 for the CO3D data, DINO-CoSeg [47] exhibits limited performance in terms of background confusion among foreground objects. The prediction of SAM [32] provides fine-grained boundary information but is noisy and lacks the detection of segmentation boundaries. Our method achieves high-quality geometric and textural consistent foreground maps without inducing noise, e.g., it can recover the complex structures of the bicycle frame and render detailed textures in *Bicycle* example scene.

Fig. 6 shows comparisons on BlendedMVS dataset [26]. From the visualizations, we see that Surface-SOS produces more view-consistent masks than other NeRF-based methods. It even generates finer details than supervised Semantic-NeRF

Fig. 8. More visualizations details of mask and RGB rendering results at the high-resolution of 3840×2160.

and SAM. For example, in the first row, Surface-SOS can distinguish the openwork details adjacent to the sleeve, and yield accurate segmentation of the shoelace in the bottom row.

Another comparison of object-centric scenes is shown in Fig. 7, these examples span a wide range of human and non-human species on the complex scene, demonstrating the superiority of our proposed methods for the geometrical consistent foreground mask, as well as its cross-view correspondence appearance of foreground and background.

The more visualizations details of mask and RGB rendering results at the high-resolution of 3840×2160 are shown in Fig. 8. It turns out that Surface-SOS can produce finer details as the input resolution increases.

2) Comparisons on Dynamic Scenes: The above results indicated that Surface-SOS can achieve promising segmentation quality on static scenes. To further evaluate the effectiveness on real scenes, we provide a comparison on our casually-captured videos (i.e., scene *Cat* and the custom stereo video {*Texting* and *Kevin*} obtained from [29]. Fig. 9 presents examples of the qualitative comparison of our method against SAM [32], DINO-CoSeg [47], RVM [9] and the untrained network-based methods LNA [42] for video layers decomposition. RVM gets excessive smoothing and blurry edges occurred in the high-frequency appearance on edges and textures. LNA outputs the nonsensical group separations due to motion signals that may be uninformative or even dishonest in cases such as deformable objects and objects with moderate motion. As shown in Fig. 9 (f), our method successfully decomposes the specifying ambiguous foreground and background with two complementary SDF-based representation modules, which is sufficient to obtain visually satisfying results. This suggests that the geometry and appearance cues in forward-backward frames can benefit object segmentation with different viewpoints consistency. Therefore, our method supports video decomposition containing moderate object motion. We encourage readers to review our supplemental videos for a dynamic visualization of qualitative results.

C. Quantitative Results

The quantitative results of compared approaches on four benchmark datasets as well as our captured data (scene *Dance*) are presented in Table I. From the results we can see that, our method outperforms supervised 2D object segmentation methods and the supervised NeRF-based segmentation method (i.e., Semantic-NeRF [46]). CO3D provides coarse segmentation maps using PointRend [31] while parts of the annotations are missing. Among self-supervised learning frameworks, Surface-SOS performs on par in both evaluation metrics and visualization for view consistency.

TABLE I
QUANTITATIVE EVALUATION OF OBJECT SEGMENTATION ON THE STATIC SCENES. THE BEST RESULTS ARE MARKED IN **Bold Font**

Dataset LLFF [25]	SAD ↓	MSE ↓	mIoU ↑	Acc. ↑
Mask-RCNN [7]	-	-	-	-
SAM [32] (Mask Init.)	10.386	0.238	0.767	0.891
DINO-CoSeg [47]	9.388	0.208	0.628	0.787
Semantic-NeRF [21]	8.736	0.185	0.897	0.918
NeRF-SOS [52]	9.172	0.191	0.865	0.857
RFP [53]	9.494	0.229	0.782	0.829
Surface-SOS (ours)	**8.655**	**0.181**	**0.903**	**0.918**
Dataset CO3D [27]	SAD ↓	MSE ↓	mIoU ↑	Acc. ↑
Mask-RCNN [7] (Mask Init.)	4.021	0.296	0.865	0.929
SAM [32]	3.265	0.226	0.876	0.940
DINO-CoSeg [47]	3.990	0.297	0.835	0.910
Semantic-NeRF [21]	3.534	0.272	0.851	0.924
NeRF-SOS [52]	3.588	0.275	0.844	0.915
Surface-SOS (ours)	**3.011**	**0.218**	**0.883**	**0.945**
Dataset BlendedMVS [26]	SAD ↓	MSE ↓	mIoU ↑	Acc. ↑
Mask-RCNN [7] (Mask Init.)	-	-	-	-
SAM [32]	6.963	0.165	0.929	0.936
DINO-CoSeg [47]	7.861	0.169	0.910	0.922
Semantic-NeRF [21]	7.915	0.191	**0.935**	**0.955**
NeRF-SOS [52]	7.739	0.192	0.924	0.934
Surface-SOS (ours)	**6.872**	**0.146**	0.931	0.941
Dataset TUM [28]	SAD ↓	MSE ↓	mIoU ↑	Acc. ↑
Mask-RCNN [7] (Mask Init.)	14.203	0.532	0.845	0.969
SAM [32]	13.108	0.576	0.864	0.981
DINO-CoSeg [47]	12.949	0.546	0.821	0.966
Semantic-NeRF [21]	13.040	0.532	0.869	0.980
NeRF-SOS [52]	12.711	0.496	0.843	0.975
Surface-SOS (ours)	**9.138**	**0.402**	**0.870**	**0.989**
Scene *Dance*	SAD ↓	MSE ↓	mIoU ↑	Acc. ↑
Mask-RCNN [7] (Mask Init.)	8.066	0.490	0.726	0.820
SAM [32]	6.803	0.380	0.886	0.840
DINO-CoSeg [47]	6.893	0.384	0.843	0.887
Semantic-NeRF [21]	7.707	0.439	0.924	0.934
NeRF-SOS [52]	7.021	0.393	0.916	0.928
Surface-SOS (ours)	**6.015**	**0.355**	**0.935**	**0.942**

D. Ablation Studies

To achieve a high-quality surface representation and analyze the correlation between the object surface representation and object segmentation, we present the performance with different design choices. These include variations with and without a coarse mutilated mask input, with a coarse mask, and with or without our proposed FoCoR and BaCo modules, as well as the inclusion of the sparsity loss. The corresponding performances are reported in Table II and Fig. 10. These results convey several observations.

Firstly, Surface-SOS can effectively recover dense 3D surface structures from multi-view images even without auxiliary

Fig. 9. Visual comparisons on the dynamic scenes *Texting*, *Kevin* and *Cat*. RVM [9] gets excessive smoothing and blurry edges occurring in the high-frequency appearance information on edges and textures. LNA [42] outputs the nonsensical group separations due to motion signals that may be uninformative or even dishonest in cases such as deformable objects and objects with moderate motion. Our method successfully decomposes temporally and geometrically consistent foreground, as well as textural, complete background.

TABLE II

RESULTS IN ABLATION STUDY

	Teddy				*Cat*			
	SAD ↓	MSE ↓	mIoU ↑	Acc. ↑	SAD ↓	MSE ↓	mIoU ↑	Acc. ↑
w/o mask Init. + w/ FoCoR + w/o BaCo	16.828	0.356	0.875	0.851	10.603	0.586	0.761	0.724
w/o mask Init. + w/ FoCoR + w/ BaCo	14.188	0.517	0.857	0.899	8.066	0.490	0.726	0.820
w/ mask Init. + w/o FoCoR + w/o BaCo	14.828	0.496	0.875	0.891	8.215	0.555	0.775	0.846
w/ mask Init. + w/ FoCoR + w/o BaCo	11.476	0.419	0.914	0.938	6.892	0.383	0.871	0.887
w/ mask Init. + w/ FoCoR + w/ BaCo + w/o sparsity	9.956	0.407	0.938	0.939	4.192	0.383	0.926	0.961
Surface-SOS (ours)	**8.685**	**0.328**	**0.950**	**0.961**	**3.434**	**0.304**	**0.946**	**0.975**

TABLE III

QUANTITATIVE COMPARISON ON DIFFERENT SINGLE-VIEW OBJECT SEGMENTATION METHODS. THE BEST RESULTS ARE MARKED IN **Bold Font**

Scene *Texting*	SAD ↓	MSE ↓	mIoU ↑	Acc.↑
Surface-SOS (w/o mask Init.)	18.559 (-)	0.386 (-)	0.825 (-)	0.918 (-)
Mask-RCNN [7]	19.588	0.390	0.873	0.921
Surface-SOS (w/ Mask-RCNN)	18.316 (↓ 1.272)	0.377 (↓ 0.013)	0.900 (↑ 0.027)	0.933 (↑ 0.012)
SAM [32]	15.598	0.347	0.955	0.938
Surface-SOS (w/ SAM)	**14.828 (↓ 0.770)**	**0.326 (↓ 0.021)**	**0.965 (↑ 0.010)**	**0.951 (↑ 0.013)**
RVM [9]	18.482	0.417	0.883	0.876
Surface-SOS (w/ RVM)	16.828 (↓ 1.654)	0.356 (↓ 0.061)	0.895 (↑ 0.012)	0.891 (↑ 0.015)
Scene *Kevin*	SAD ↓	MSE ↓	mIoU ↑	Acc.↑
Surface-SOS (w/o mask Init.)	13.336 (-)	0.396 (-)	0.836 (-)	0.941 (-)
Mask-RCNN [7]	16.012	0.417	0.824	0.872
Surface-SOS (w/ Mask-RCNN)	14.702 (↓ 1.311)	0.411 (↓ 0.006)	0.846 (↑ 0.022)	0.879 (↑ 0.007)
SAM [32]	10.603	0.386	0.861	0.924
Surface-SOS (w/ SAM)	9.593 (↓ 1.010)	0.356 (↓ 0.030)	0.879 (↑ 0.018)	0.932 (↑ 0.008)
RVM [9]	11.366	0.394	0.860	0.960
Surface-SOS (w/ RVM)	**8.965 (↓ 2.401)**	**0.362 (↓ 0.032)**	**0.881 (↑ 0.021)**	**0.974 (↑ 0.014)**

inputs of object masks. As we can see in Fig. 10(a), this approach results in a reasonable foreground and background decomposition.

Secondly, when providing the SDF-based surface representation (i.e., FoCoR module) with coarse masks, the network learns 3D geometry implicitly and generates an accurate

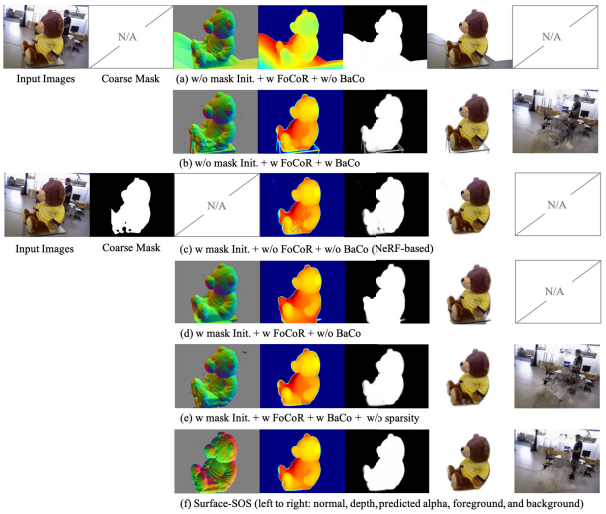

Fig. 10. Ablation Studies. Compare Surface-SOS to different design choices: with and without a coarse mutilated mask input, with a coarse mask and with or without our proposed FoCoR and BaCo modules, as well as the sparsity loss. Without the coarse mask initiation, Surface-SOS can decompose reasonable foreground and background, part of the desk is segmented out due to the view-consistent geometry for the static foreground. When providing the FoCoR module with coarse masks, the network is able to learn 3D geometry implicitly and generate an accurate foreground decomposition, By adding the background learning module (i.e., BaCo) and sparsity loss to the SDF-based surface representation, the resulting geometric surfaces become more compact and prevent holes in the object alpha matte.

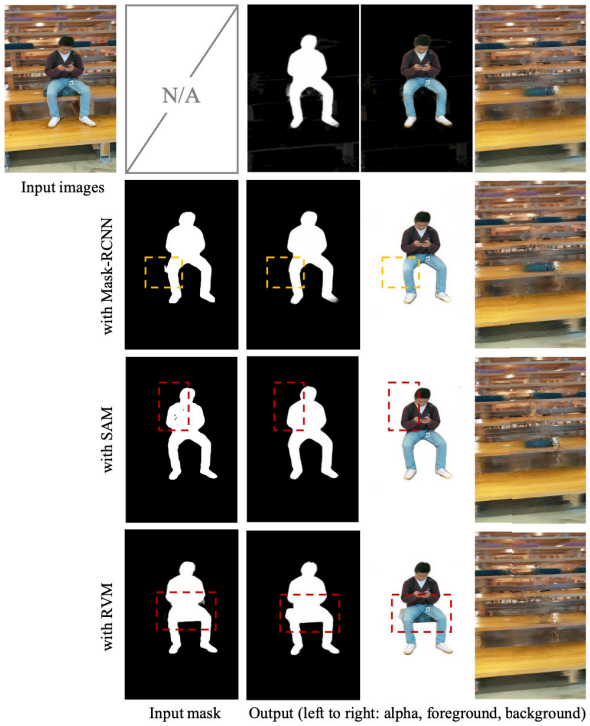

Fig. 11. Impact of the mask initialization. Surface-SOS can effectively decompose a complex scene into foreground and background without initiating a rough mask. By introducing coarse segmentation masks as additional input, Surface-SOS is able to refine single-view segmentation, such as Mask-RCNN [7], SAM [32], and RVM [9]. For instance, compared to the coarse masks of Mask-RCNN and SAM, the initial mask of RVM contains incomplete cushions, Surface-SOS can effectively recover dense 3D surface structures from multi-view images and produce high-quality segmentation maps. These examples show that even rough segmentation results in this step can yield high-quality foreground at the end, making the system practically applicable.

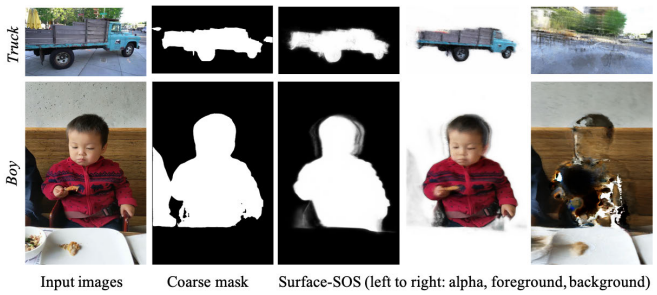

Fig. 12. Failure cases. On the unbounded scene *Truck* from hand-held 360 capture Tank and Temples dataset [60], the generated results are blurry and lack fine details. On the custom stereo video *Boy* from [29], it mistakenly matches several discrete pixel patches.

foreground decomposition. Specifically, the background learning module (i.e., BaCo) prevents the occurrence of uncontrollable free surfaces, and the sparsity loss encourages the model to render images with the minimum content required for recovery and prevents the duplication of foreground representations, which further promotes the creation of an accurate foreground and background decomposition with fine detail. These examples demonstrate the benefits of accurately predicting object geometry using two complementary neural representations for self-supervised object segmentation.

Moreover, by introducing coarse masks as additional input, Surface-SOS is able to refine the segmentation remarkably. In Fig. 11 we present an analysis of mask initialization in our framework by removing the coarse mask input, as well as applying several rough segmentation acquired by different single-view methods [7], [9], [32]. Extensive experiments are presented in Table III, and the results clearly demonstrate that Surface-SOS outperforms all of the original single-view methods by a large margin. For instance, for the scenes *Kevin*, in terms of SAD, MSE, mIoU, and Acc., Surface-SOS with the RVM masks initialization surpasses the RVM by -2.401, -0.032, 0.021 and 0.014, respectively. These examples show that even rough segmentation results in this step can yield high-quality foreground at the end, making the system practically applicable.

V. Conclusion

In this paper, we present Surface-SOS, a new self-supervised learning framework for delicate segmentation from multi-view images that are geometrically consistent. To leverage 3D object-level geometry and 2D image appearance cues of the one-to-one dense mapping in 3D space, we designed a special neural scene decomposition approach containing two complementary neural representation modules, i.e. FoCoR and BaCo, processing the foreground and background, respectively. In this manner, we can effectively decompose scenes into foreground and background, including its convincing segmentation maps. Our framework can be implemented to refine 2D single-view object segmentation on complex scenes with only unlabeled multi-view images. Extensive experiments on various multi-view benchmark datasets and monocular stereo videos validated the effectiveness of the Surface-SOS,

significantly improving the supervised 2D single-view object segmentation results, and generating finer-grained segmentation than existing multi-view NeRF-based frameworks.

Though promising, there are still some limitations and drawbacks of the proposed method. As we extract geometric constraints by leveraging the SDF-based surfaces representation from a sparse set of images, it cannot segment across scenes. Furthermore, it faces more challenges on unbounded scenes, such as the hand-held 360 capture large-scale Tank and Temples datasets [60], due to the lack of solid geometry in the scenes. Our method supports videos containing moderate object motion. It breaks for extreme object motion. Some failure cases are shown in Fig. 12. Integrating our approach with learning-based pose estimation and neural dynamic representation of large motion and deformed objects is an interesting future direction.

REFERENCES

[1] A. Djelouah, J.-S. Franco, E. Boyer, F. Le Clerc, and P. Pérez, "Multi-view object segmentation in space and time," in *Proc. IEEE Int. Conf. Comput. Vis.*, Oct. 2013, pp. 2640–2647.

[2] T. Zhou, J. Li, S. Wang, R. Tao, and J. Shen, "MATNet: Motion-attentive transition network for zero-shot video object segmentation," *IEEE Trans. Image Process.*, vol. 29, pp. 8326–8338, 2020.

[3] J. Zhang et al., "Editable free-viewpoint video using a layered neural representation," *ACM Trans. Graph.*, vol. 40, no. 4, pp. 1–18, Aug. 2021.

[4] V. Lazova, V. Guzov, K. Olszewski, S. Tulyakov, and G. Pons-Moll, "Control-NeRF: Editable feature volumes for scene rendering and manipulation," in *Proc. IEEE/CVF Winter Conf. Appl. Comput. Vis. (WACV)*, Jan. 2023, pp. 4329–4339.

[5] B. Yang et al., "Learning object-compositional neural radiance field for editable scene rendering," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 13779–13788.

[6] Y. Cai, X. Li, Y. Wang, and R. Wang, "An overview of panoramic video projection schemes in the IEEE 1857.9 standard for immersive visual content coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 32, no. 9, pp. 6400–6413, Sep. 2022.

[7] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2961–2969.

[8] L. Jing, Y. Chen, and Y. Tian, "Coarse-to-fine semantic segmentation from image-level labels," *IEEE Trans. Image Process.*, vol. 29, pp. 225–236, 2020.

[9] S. Lin, L. Yang, I. Saleemi, and S. Sengupta, "Robust high-resolution video matting with temporal guidance," in *Proc. IEEE/CVF Winter Conf. Appl. Comput. Vis. (WACV)*, Jan. 2022, pp. 3132–3141.

[10] E. Lu, F. Cole, T. Dekel, A. Zisserman, W. T. Freeman, and M. Rubinstein, "Omnimatte: Associating objects and their effects in video," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 4505–4513.

[11] E. Lu et al., "Layered neural rendering for retiming people in video," 2020, *arXiv:2009.07833*.

[12] C. Wu, "Towards linear-time incremental structure from motion," in *Proc. Int. Conf. 3D Vis.*, Jun. 2013, pp. 127–134.

[13] J.-Y. Guillemaut and A. Hilton, "Joint multi-layer segmentation and reconstruction for free-viewpoint video applications," *Int. J. Comput. Vis.*, vol. 93, no. 1, pp. 73–100, May 2011.

[14] E. T. Hall, "A system for the notation of proxemic behavior," *Amer. Anthropologist*, vol. 65, no. 5, pp. 1003–1026, Oct. 1963.

[15] B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng, "NeRF: Representing scenes as neural radiance fields for view synthesis," *Commun. ACM*, vol. 65, no. 1, p. 99, Dec. 2021.

[16] L. Liu, J. Gu, K. Z. Lin, T.-S. Chua, and C. Theobalt, "Neural sparse voxel fields," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 33, 2020, pp. 15651–15663.

[17] T. Müller, A. Evans, C. Schied, and A. Keller, "Instant neural graphics primitives with a multiresolution hash encoding," *ACM Trans. Graph.*, vol. 41, no. 4, pp. 1–15, Jul. 2022.

[18] Y. Jiang, D. Ji, Z. Han, and M. Zwicker, "SDFDiff: Differentiable rendering of signed distance fields for 3D shape optimization," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 1248–1258.

[19] P. Wang, L. Liu, Y. Liu, C. Theobalt, T. Komura, and W. Wang, "NeuS: Learning neural implicit surfaces by volume rendering for multi-view reconstruction," in *Proc. NIPS*, 2021, pp. 1–23.

[20] Y. Wang, Q. Han, M. Habermann, K. Daniilidis, C. Theobalt, and L. Liu, "NeuS2: Fast learning of neural implicit surfaces for multi-view reconstruction," 2022, *arXiv:2212.05231*.

[21] S. Zhi, T. Laidlow, S. Leutenegger, and A. J. Davison, "In-place scene labelling and understanding with implicit scene representation," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, Oct. 2021, pp. 15838–15847.

[22] W. Zhang, R. Xing, Y. Zeng, Y.-S. Liu, K. Shi, and Z. Han, "Fast learning radiance fields by shooting much fewer rays," *IEEE Trans. Image Process.*, vol. 32, pp. 2703–2718, 2023.

[23] J. Philip and V. Deschaintre, "Floaters no more: Radiance field gradient scaling for improved near-camera training," in *Eurographics Symposium on Rendering*, T. Ritschel and A. Weidlich, Eds. Eindhoven, The Netherlands: The Eurographics Association, 2023, pp. 25–35, doi: 10.2312/sr.20231122.

[24] L. Yariv, J. Gu, Y. Kasten, and Y. Lipman, "Volume rendering of neural implicit surfaces," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 34, Dec. 2021, pp. 4805–4815.

[25] B. Mildenhall et al., "Local light field fusion: Practical view synthesis with prescriptive sampling guidelines," *ACM Trans. Graph.*, vol. 38, no. 4, pp. 1–14, Jul. 2019.

[26] Y. Yao et al., "Blendedmvs: A large-scale dataset for generalized multi-view stereo networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2020, pp. 1790–1799.

[27] J. Reizenstein, R. Shapovalov, P. Henzler, L. Sbordone, P. Labatut, and D. Novotny, "Common objects in 3D: Large-scale learning and evaluation of real-life 3D category reconstruction," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, Oct. 2021, pp. 10901–10911.

[28] J. Sturm, N. Engelhard, F. Endres, W. Burgard, and D. Cremers, "A benchmark for the evaluation of RGB-D SLAM systems," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Oct. 2012, pp. 573–580.

[29] X. Luo, J.-B. Huang, R. Szeliski, K. Matzen, and J. Kopf, "Consistent video depth estimation," *ACM Trans. Graph.*, vol. 39, no. 4, pp. 1–71, Aug. 2020.

[30] J. L. Schonberger and J.-M. Frahm, "Structure-from-motion revisited," in *Proc. CVPR*, 2016, pp. 4104–4113.

[31] A. Kirillov, Y. Wu, K. He, and R. Girshick, "PointRend: Image segmentation as rendering," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 9799–9808.

[32] A. Kirillov et al., "Segment anything," 2023, *arXiv:2304.02643*.

[33] H. Li, G. Chen, G. Li, and Y. Yu, "Motion guided attention for video salient object detection," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 7273–7282.

[34] B. Jiang, Z. Zhou, X. Wang, J. Tang, and B. Luo, "CmSalGAN: RGB-D salient object detection with cross-view generative adversarial networks," *IEEE Trans. Multimedia*, vol. 23, pp. 1343–1353, 2021.

[35] T. Zhou, J. Li, X. Li, and L. Shao, "Target-aware object discovery and association for unsupervised video multi-object segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2021, pp. 6985–6994.

[36] T. Zhuo, Z. Cheng, P. Zhang, Y. Wong, and M. Kankanhalli, "Unsupervised online video object segmentation with motion property understanding," *IEEE Trans. Image Process.*, vol. 29, pp. 237–249, 2020.

[37] T. Porter and T. Duff, "Compositing digital images," in *Proc. 11th Annu. Conf. Comput. Graph. Interact. Techn.*, Jan. 1984, pp. 253–259.

[38] Z. Fan, J. Lu, C. Wei, H. Huang, X. Cai, and X. Chen, "A hierarchical image matting model for blood vessel segmentation in fundus images," *IEEE Trans. Image Process.*, vol. 28, no. 5, pp. 2367–2377, May 2019.

[39] X. Fang, S.-H. Zhang, T. Chen, X. Wu, A. Shamir, and S.-M. Hu, "User-guided deep human image matting using arbitrary trimaps," *IEEE Trans. Image Process.*, vol. 31, pp. 2040–2052, 2022.

[40] J. Sun, J. Jia, C.-K. Tang, and H.-Y. Shum, "Poisson matting," in *Proc. ACM SIGGRAPH Papers*, 2004, pp. 315–321.

[41] S. Lin, A. Ryabtsev, S. Sengupta, B. L. Curless, S. M. Seitz, and I. Kemelmacher-Shlizerman, "Real-time high-resolution background matting," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2021, pp. 8762–8771.

[42] Y. Kasten, D. Ofri, O. Wang, and T. Dekel, "Layered neural atlases for consistent video editing," *ACM Trans. Graph.*, vol. 40, no. 6, pp. 1–12, Dec. 2021.

[43] C. Zhang, G. Li, G. Lin, Q. Wu, and R. Yao, "CycleSegNet: Object co-segmentation with cycle refinement and region correspondence," *IEEE Trans. Image Process.*, vol. 30, pp. 5652–5664, 2021.
[44] Z. Yuan, T. Lu, and Y. Wu, "Deep-dense conditional random fields for object co-segmentation," in *Proc. 26th Int. Joint Conf. Artif. Intell.*, Aug. 2017, pp. 3371–3377.
[45] W. Li, O. Hosseini Jafari, and C. Rother, "Deep object co-segmentation," in *Computer Vision—ACCV*. Cham, Switzerland: Springer, 2019, pp. 638–653.
[46] H. Chen, Y. Huang, and H. Nakayama, "Semantic aware attention based deep object co-segmentation,' in *Computer Vision—ACCV*. Cham, Switzerland: Springer, 2019, pp. 435–450.
[47] S. Amir, Y. Gandelsman, S. Bagon, and T. Dekel, "Deep ViT features as dense visual descriptors," 2021, *arXiv:2112.05814*.
[48] I. Kompatsiaris, D. Tzovaras, and M. G. Strintzis, "3D model-based segmentation of videoconference image sequences," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 8, no. 5, pp. 547–561, Sep. 1998.
[49] G. Gordon, T. Darrell, M. Harville, and J. Woodfill, "Background estimation and removal based on range and color," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, vol. 2, Jun. 1999, pp. 459–464.
[50] Z. Gang and Q. Long, "Silhouette extraction from multiple images of an unknown background," in *Proc. Asian Conf. Comput. Vis.*, 2004, pp. 1–6.
[51] W. Lee, W. Woo, and E. Boyer, "Silhouette segmentation in multiple views," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 7, pp. 1429–1441, Jul. 2011.
[52] Z. Fan, P. Wang, Y. Jiang, X. Gong, D. Xu, and Z. Wang, "NeRF-SOS: Any-view self-supervised object segmentation on complex scenes," 2022, *arXiv:2209.08776*.
[53] X. Liu, J. Chen, H. Yu, Y.-W. Tai, and C.-K. Tang, "Unsupervised multi-view object segmentation using radiance field propagation," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 35, 2022, pp. 17730–17743.
[54] J. T. Barron, B. Mildenhall, D. Verbin, P. P. Srinivasan, and P. Hedman, "Mip-NeRF 360: Unbounded anti-aliased neural radiance fields," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2022, pp. 5470–5479.
[55] L. Mescheder, M. Oechsle, M. Niemeyer, S. Nowozin, and A. Geiger, "Occupancy networks: Learning 3D reconstruction in function space," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2019, pp. 4460–4470.
[56] J. J. Park, P. Florence, J. Straub, R. Newcombe, and S. Lovegrove, "DeepSDF: Learning continuous signed distance functions for shape representation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2019, pp. 165–174.
[57] A. Gropp, L. Yariv, N. Haim, M. Atzmon, and Y. Lipman, "Implicit geometric regularization for learning shapes," in *Proc. Mach. Learn. Syst.*, 2020, pp. 3569–3579.
[58] X. Long, C. Lin, P. Wang, T. Komura, and W. Wang, "SparseNeuS: Fast generalizable neural surface reconstruction from sparse views," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2022, pp. 210–227.
[59] P. Savarese, S. S. Y. Kim, M. Maire, G. Shakhnarovich, and D. McAllester, "Information-theoretic segmentation by inpainting error maximization," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 4028–4038.
[60] G. Riegler and V. Koltun, "Free view synthesis," in *Computer Vision—ECCV*, Glasgow, U.K. Cham, Switzerland: Springer, 2020, pp. 623–640.

Xiaoyun Zheng received the M.S. degree in mechanical engineering from Tongji University, Shanghai, China, in 2019. She is currently pursuing the Ph.D. degree with the School of Computer Science, Peking University Shenzhen Graduate School, Shenzhen, China. Her research interests include video/image processing, computer vision, and 3D human reconstruction.

Liwei Liao received the M.S. degree in micro electronics from Peking University, Beijing China, in 2019. He is currently pursuing the Ph.D. degree in computer science with Peking University Shenzhen Graduate School, Shenzhen, China. His research interests include multi-modal machine learning and 3D human reconstruction.

Jianbo Jiao (Member, IEEE) received the Ph.D. degree in computer science from the City University of Hong Kong, in 2018. He was a Visiting Scholar with the Beckman Institute, University of Illinois at Urbana–Champaign, from 2017 to 2018. He is currently an Assistant Professor with the School of Computer Science, University of Birmingham, a Royal Society Short Industry Fellow, and a Visiting Researcher with the University of Oxford, U.K. Before joining Birmingham, he was a Postdoctoral Researcher with the Department of Engineering Science, University of Oxford. His research interests include machine learning and computer vision. He was a recipient of Hong Kong Ph.D. Fellowship Scheme (HKPFS).

Feng Gao (Member, IEEE) received the B.S. degree in computer science from University College London, in 2007, and the Ph.D. degree in computer science from Peking University, in 2018. He was a Postdoctoral Research Fellow with The Future Laboratory, Tsinghua University, from 2018 to 2020. He has been with Peking University, as an Assistant Professor, since 2020. His research interests include the intersection of computer science and art, including but not limit on artificial intelligence and painting art, deep learning, and painting robot.

Ronggang Wang (Member, IEEE) received the Ph.D. degree from the Institute of Computing Technology, Chinese Academy of Sciences, in 2006. He is currently a Professor with the School of Electronic and Computer Engineering, Peking University Shenzhen Graduate School. He has made over 100 technical contributions to ISO/IEC MPEG, IEEE 1857, and China AVS. He has authored more than 150 articles and held more than 100 patents. His research interests include immersive video coding and processing. He has been serving as the IEEE 1857.9 Immersive Video Coding Standard Sub-Group Chair and has been China AVS Virtual Reality Sub-Group Chair since 2016.