Free-Moving Object Reconstruction and Pose Estimation with Virtual Camera

Haixin Shi[1], **Yinlin Hu**[2], **Daniel Koguciuk**[2], **Juan-Ting Lin**[2]
Mathieu Salzmann[1], **David Ferstl**[2]

[1]EPFL
[2]Magic Leap

Abstract

We propose an approach for reconstructing free-moving object from a monocular RGB video. Most existing methods either assume scene prior, hand pose prior, object category pose prior, or rely on local optimization with multiple sequence segments. We propose a method that allows free interaction with the object in front of a moving camera without relying on any prior, and optimizes the sequence globally without any segments. We progressively optimize the object shape and pose simultaneously based on an implicit neural representation. A key aspect of our method is a virtual camera system that reduces the search space of the optimization significantly. We evaluate our method on the standard HO3D dataset and a collection of egocentric RGB sequences captured with a head-mounted device. We demonstrate that our approach outperforms most methods significantly, and is on par with recent techniques that assume prior information.

Project Page — https://haixinshi.github.io/fmov/

1 Introduction

Understanding 3D objects around us is a fundamental problem in computer vision, and also a critical component in many applications, such as augmented reality (AR) (Billinghurst 2021) and robot manipulation (Thalhammer et al. 2023). This requires an accurate reconstruction and pose estimation of such objects. With a monocular RGB camera, most current work tackle this problem with major simplifications, either by moving the camera around a static object (Oechsle, Peng, and Geiger 2021; Rünz et al. 2020; Yariv et al. 2020; Zou et al. 2024) or by rotating the object with hands in front of a stationary camera (Rusinkiewicz, Hall-Holt, and Levoy 2002; Tzionas and Gall 2015; Weise, Leibe, and Van Gool 2008; Weise et al. 2009; Chen et al. 2023).

In this paper, we investigate a more general setting with the example of an AR device where the object is free-moving in front of a head-mounted camera. We neither assume any object category prior nor any hand prior in this new setting, which allows the objects to be moved in any manner, or freely manipulated with any grasping style if moved by hands.

Copyright © 2025, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

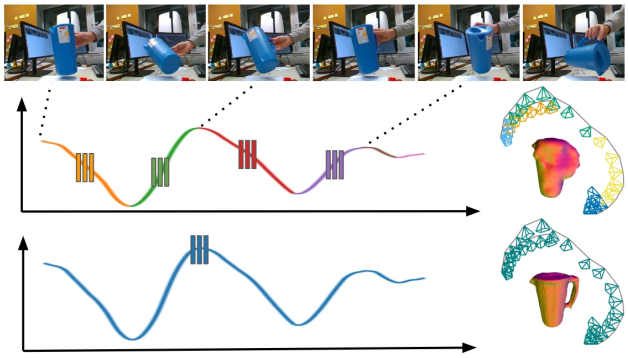


Figure 1: Free-moving object reconstruction and pose estimation from a monocular RGB video. Top: Input sequence. Middle: Existing methods (Hampali et al. 2023) rely on segment-wise optimization with multiple easy segments of the sequence, which tends to be local optimal, as shown in different trajectory colors. Bottom: Our method optimizes object shape and pose progressively without any segments, producing globally consistent shape and pose results.

Without pose initialization, most recent methods (Lin et al. 2021; Jeong et al. 2021; Rosinol, Leonard, and Carlone 2022; Rozumnyi et al. 2023) optimize the shape representation and poses simultaneously. However, most of them either rely on geometry clues of the background, which is inapplicable for free-moving objects, or can only handle a restricted range of viewpoints. On the other hand, some recent methods (Hampali et al. 2023) use progressive training to solve this problem and rely on segment-wise optimization based on multiple overlapping segments of the sequence. This strategy, however, suffers mainly in two ways. First, the frame selection of each segment is error-prone, since it typically relies on the changes of mask area of the target between consecutive frames, which can hardly be generalized to different object shapes. Second, the segment-wise optimization is inherently local and suboptimal, as shown in Fig. 1.

We propose a method for joint reconstruction and pose estimation of free-moving objects without any segments, which can be globally optimized with a single network. Our observation is that the unknown pose trajectory of the object can be simplified with a new virtual camera system that al-

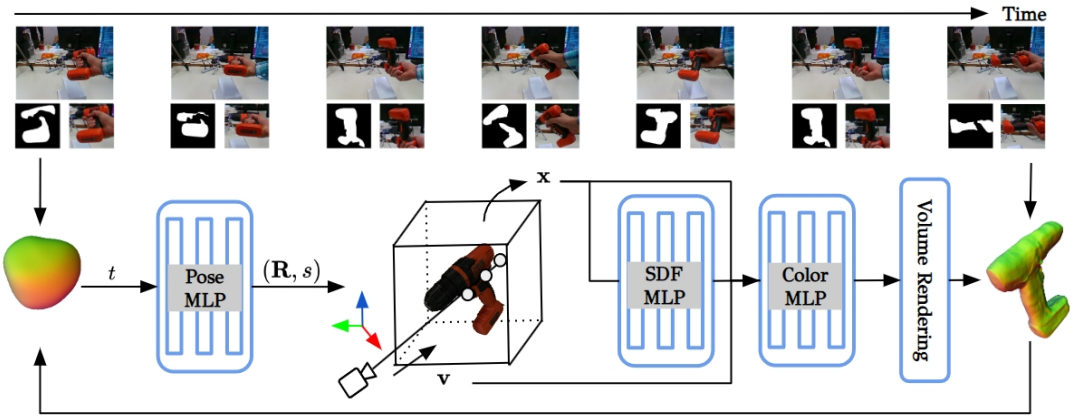


Figure 2: Overview of our method. We first use off-the-shelf 2D segmentation methods to get object mask in each frame, and then optimize MLP networks w.r.t. a virtual camera system, with which the camera always points to areas near the object center, as illustrated as colored 3D axis in the figure. We optimize three MLPs with progressively added images. For each frame with time index t, we use Pose MLP to predict the object pose $(\mathbf{R}, s)$, which corresponds to the rotation and the distance from the camera center to the object center, summing up to only 4 degrees of freedom. For each 3D point $\mathbf{x}$ along the view direction $\mathbf{v}$, we use SDF MLP and Color MLP to predict its corresponding SDF value and color opacity, respectively. We compare the rendered image with the input and update MLP networks based on volume rendering. We finally conduct a virtual-to-real conversion and refine all the results w.r.t. the real camera.

ways points to the object center with the guidance of 2D object masks, which reduces the search space of the optimization significantly. We first use off-the-shelf 2D segmentation methods to get object masks in each frame, and then optimize the network w.r.t. the virtual camera. To handle the approximation error between virtual camera and real camera, we finally convert the results to real camera coordinate system and refine all the results w.r.t. the real camera. We evaluate our method on both the HO3D dataset (Hampali et al. 2020) with fixed camera and a collection of data captured using a head-mounted AR device with egocentric views. The experiments show that our method outperforms traditional methods and most baselines significantly.

We summarize our contributions as follows. First, we investigate the problem of existing methods in reconstructing free-moving object from a RGB monocular video. Second, we propose a simple-but-effective strategy with a virtual camera system that simplifies the object trajectory and reduce the search space of the optimization significantly. Finally, we demonstrate the effectiveness of our method on datasets with either a fixed or egocentric camera.

2 Related Work

3D Reconstruction is a fundamental problem in computer vision. Traditional methods (Agarwal et al. 2011; Mohr, Quan, and Veillon 1995; Pollefeys et al. 2004; Snavely, Seitz, and Szeliski 2006), typically COLMAP (Schönberger and Frahm 2016), first estimate camera parameters with 2D matching and then reconstruct the scene with multi-view-stereo (MVS) techniques (Cheng et al. 2021). Most recent methods solve the reconstruction by optimizing a neural implicit representation with rendering techniques (Mildenhall et al. 2020; Pumarola et al. 2021; Wang et al. 2021b, 2023). Although they achieve high-quality reconstruction results, most of them rely on SfM to obtain accurate camera poses for each frame. However, most SfM-based methods assume static rigid scenes and only work when there are enough textures in the scene (Schönberger and Frahm 2016; Sweeney 2016; Snavely 2011), which is inapplicable in our setting where the object moves independently of the background and the object is often texture-less. Some recent works try to remove the SfM pose initialization by optimizing the neural representation and camera poses simultaneously (Lin et al. 2021; Jeong et al. 2021; Rosinol, Leonard, and Carlone 2022; Bian et al. 2023; Sabae, Baraka, and Hadhoud 2023). However, they can only work with forward-facing scenarios with a restricted range of view. The recent method (Hampali et al. 2023) tries to divide the whole sequence into multiple easy short segments to facilitate the optimization. However, the frame selection of the segmenting procedure is shape-dependent and the segment-wise based optimization is local optimal. By contrast, our method is segment-free and can produce globally consistent results.

Object Pose Estimation aims to produce accurate 3D rotation and 3D translation of the object w.r.t. the camera, which usually serves as pose initialization for dynamic object reconstruction. Most recent object pose methods (Hodan et al. 2018; Hu, Fua, and Salzmann 2022; Hu et al. 2021; Su et al. 2022; Wang et al. 2021a) first establish 3D-to-2D correspondences via networks and then use a Perspective-n-Points (PnP) solver (Lepetit, Moreno-Noguer, and Fua 2009; Moreno-Noguer, Lepetit, and Fua 2007) to get the pose results. However, most of them rely on the object's 3D mesh, which is one of the goals of reconstruction and is inapplicable in this work. Some recent category-level methods (Chen et al. 2020; Tian, Ang, and Lee 2020; Weng et al. 2021; Yu,

Zhai, and Xia 2024; Peng et al. 2022) do not rely on the target's mesh explicitly by training the network with mixed data from different instances of the same category. However, they are limited to known categories in the training set and cannot be generalized. By contrast, we do not rely on any category prior of the object but optimize the shape, color, and pose of the target simultaneously. On the other hand, most hand-held reconstruction methods (Huang et al. 2022; Fan et al. 2024; Jiang et al. 2024) rely on the hand-object interaction, and most of them assume a firmly grasping of the object during the whole capture to leverage hand pose estimators for pose initialization. Differently, our method does not make any assumptions about specific grasping styles and is effective for free-moving objects.

3 Approach

The goal of our method is to reconstruct a rigid dynamic object from a sequence of RGB images captured with a calibrated camera. Fig. 2 shows the overview of our method.

3.1 Capture Setup and Data Pre-Processing

We capture a sequence of RGB images of dynamic objects with either a fixed or egocentric camera, where the relative pose between the camera and the object is unknown. We do not assume any object prior or any hand pose prerequisites, which allows the users to rotate the object in any grasping style or even switch between different hands during capture. Users only need to ensure the object is within the field of view of the camera, and all sides of the object are covered during capture to have a full reconstruction.

Our method relies on 2D object masks to separate the object from the background. We obtain object mask of the first frame with some simple clicks based on an interactive segmentation method (Kirillov et al. 2023), and then obtain all the object masks in the following frames with a segmentation tracking method (Cheng and Schwing 2022).

3.2 Learning Object Representation

We use the Signed Distance Function (SDF) (Wang et al. 2021b; Yariv et al. 2021; Oechsle, Peng, and Geiger 2021) as an implicit representation for the object surface, where the surface of the object is given by the zero-level set of its SDF. We learn the SDF from an image sequence, and compare the input images with volume-rendered images (Wang et al. 2021b) after converting the SDF to a radiance field. The final surface mesh is extracted by Marching Cubes (Lorensen and Cline 1987).

We follow NeuS (Wang et al. 2021b) and use MLP networks to learn the object surface and appearance:

$$(d(\mathbf{x}), \mathbf{c}(\mathbf{x}, \mathbf{v})) = F_\theta(\mathbf{x}, \mathbf{v}), \tag{1}$$

where F_θ is the MLP networks. For the 3D location $\mathbf{x}$ and viewing direction $\mathbf{v}$, F_θ predicts their SDF value $d(\mathbf{x})$ and RGB colors $\mathbf{c}(\mathbf{x}, \mathbf{v})$. In practice, we use two different MLPs to predict the surface and color field (Wang et al. 2021b), and apply positional encoding (Mildenhall et al. 2020; Müller et al. 2022) to $\mathbf{x}$ and $\mathbf{v}$ to capture high-frequency signals.

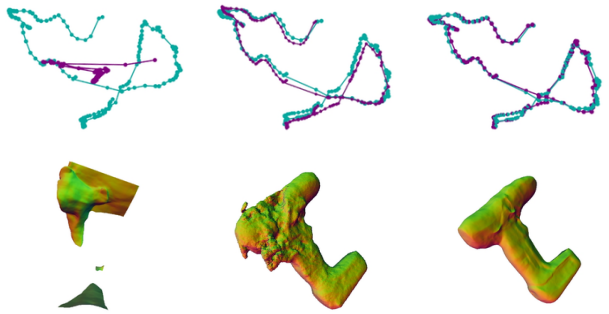
We use volume rendering (Kajiya and Von Herzen 1984) to optimize the implicit representation. The rendered color of each pixel is an integration of colors along the camera ray $\mathbf{r}$ passing through the pixel, which is usually numerically approximated in practice using quadrature (Mildenhall et al. 2020):

$$\hat{\mathbf{C}}(\mathbf{r}) = \sum_{k=1}^{N} w_k \mathbf{c}_k \tag{2}$$

with the alpha-blending coefficient $w_k = \exp(-\sum_{i=1}^{k-1} \delta_i \sigma_i)(1 - \exp(-\delta_k \sigma_k))$, where N is the number of sampled points along the ray $\mathbf{r}$, δ_k is the 3D distance between adjacent sampled points $\mathbf{x}_k$ and $\mathbf{x}_{k+1}$, and σ_k is the volume density of $\mathbf{x}_k$ after a transformation of its signed distance d_k (Wang et al. 2021b). During training, the network parameters are learned using multi-view images with photometric loss:

$$\mathcal{L}_{color} = \sum_{\mathbf{r} \in \mathcal{R}} \|\hat{\mathbf{C}}(\mathbf{r}) - \mathbf{C}(\mathbf{r})\| \tag{3}$$

to measure the difference between the rendered color $\hat{\mathbf{C}}(\mathbf{r})$ and the observed color $\mathbf{C}(\mathbf{r})$ of a pixel intersected by the ray $\mathbf{r}$, where $\mathcal{R}$ is the set of camera rays going through sampled pixels. We use the same Eikonal loss and mask loss as in NeuS for network regularization.

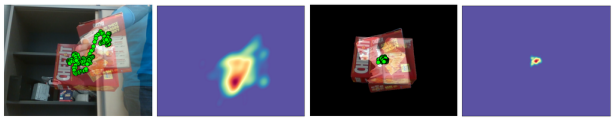


(a) BARF (b) Hampali *et al*. (c) Ours

Figure 3: Different methods for joint pose and shape optimization. (a) BARF (Lin et al. 2021) struggles in handling 360-degree sequences. (b) The segment-wise optimization of Hampali *et al*. (Hampali et al. 2023) suffers in scenarios with large pose changes. (c) Our method produces globally consistent results. We visualize ground truth pose and predicted pose in cyan and purple, respectively.

3.3 Optimization with Guided Virtual Camera

To optimize the object representation discussed in the above section, it is essential to specify both the origin and direction of camera rays or camera/object poses. We do not assume the existence of such poses in our setting. Similar to (Lin et al. 2021; Hampali et al. 2023; Bian et al. 2023; Sabae, Baraka, and Hadhoud 2023), we optimize the camera poses and the object representation simultaneously. Since camera rays are functions of camera parameters, we condition the camera rays in Eq. 3 on learnable camera poses, as illustrated as the Pose MLP in Fig. 2. We assume the camera



Real camera Virtual camera

Figure 4: Effect of virtual camera. For each camera system, the left figure shows the trajectory of the moving object, and the right figure depicts the heatmap of 2D object center across the whole HO3D dataset. The poses w.r.t the virtual camera do not have significant magnitude in both horizontal and vertical directions, which allows the poses to be approximately captured by only 4 degrees of freedom (3 for rotation and 1 for distance).

intrinsics and lens distortions of the camera are known and only optimize the camera pose in this work.

Typically, joint optimization of camera pose and radiance field can only handle forward-facing scenarios and fails to converge if the images cover a larger range of viewpoints or there are some large pose changes between consecutive images, as shown in Fig. 3.

To handle this problem, we propose to solve the optimization problem w.r.t. a new virtual camera system with the guidance of 2D object masks. Specially, given a 3D point $\mathbf{x}$ and the camera intrinsic matrix $\mathbf{K}$, we have:

$$\mathbf{u} = \mathbf{K}(\mathbf{R}\mathbf{x} + \mathbf{t}), \tag{4}$$

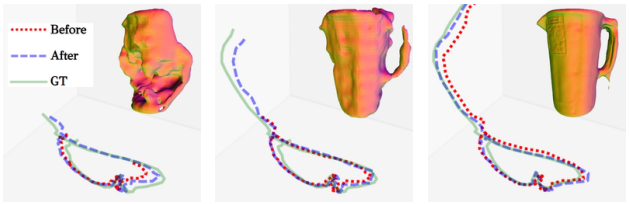
where $\mathbf{u}$ is the reprojected pixel location on the image, and $\mathbf{R}$ and $\mathbf{t}$ are the 3D rotation and 3D translation respectively. On the other hand, given the object mask obtained in Sec. 3.1, we crop the object with a transformation matrix $\mathbf{M}$, and with a virtual camera whose intrinsic matrix is $\mathbf{K}_v$, we have:

$$\begin{aligned} \mathbf{M}\mathbf{u} &= \mathbf{M}\mathbf{K}(\mathbf{R}\mathbf{x} + \mathbf{t}) \\ &= \mathbf{K}_v\mathbf{K}_v^{-1}\mathbf{M}\mathbf{K}(\mathbf{R}\mathbf{x} + \mathbf{t}) \\ &= \mathbf{K}_v(\mathbf{K}_v^{-1}\mathbf{M}\mathbf{K}\mathbf{R}\mathbf{x} + \mathbf{K}_v^{-1}\mathbf{M}\mathbf{K}\mathbf{t}), \end{aligned} \tag{5}$$

where $(\mathbf{K}_v^{-1}\mathbf{M}\mathbf{K}\mathbf{R}, \mathbf{K}_v^{-1}\mathbf{M}\mathbf{K}\mathbf{t}) \rightarrow (\mathbf{R}_v, \mathbf{t}_v)$ are the 3D rotation and 3D translation w.r.t. the new virtual camera. While note that $(\mathbf{R}_v, \mathbf{t}_v)$ is not physically-compliant and can only be approximately estimated. Instead of optimizing $(\mathbf{R}, \mathbf{t})$ directly, we optimize $(\mathbf{R}_v, \mathbf{t}_v)$ and the object representation simultaneously w.r.t. the virtual camera. Since the translation of the target w.r.t. the new virtual camera does not have significant magnitude in both horizontal and vertical directions, as illustrated in Fig. 4, we only predict the rotation and the distance from the camera center to the object center for object poses, which has only 4 degrees of freedom. This simplification of pose formulation reduces the search space of network optimization, and our experiments will show it increases the performance significantly.

3.4 Segment-Free Progressive Training

Although the virtual camera system reduces the search space of optimization, it still can not handle 360-degree sequences.



$t_1 \rightarrow t_1$+10 $t_2 \rightarrow t_2$+10 with refinement

Figure 5: Progressive training and global refinement. The first two figures show the pose and shape results of two examples during progressive training. The result improves with more images involved. The last figure shows the result with global refinement, which improves the performance further.

We use progressive training to process the images in the sequence to leverage the temporal information between consecutive images (Hampali et al. 2023; Fu et al. 2024).

We use 2D matches between different images to facilitate the optimization:

$$\begin{aligned} \mathcal{L}_{match}(\mathbf{r}_a, \mathbf{r}_b) = &\sum_{k=1}^{N}(w_{ak} \cdot \|g_b(\mathbf{x}_{ak}) - \mathbf{u}_b\|_1) \\ &+ \sum_{k=1}^{N}(w_{bk} \cdot \|g_a(\mathbf{x}_{bk}) - \mathbf{u}_a\|_1), \end{aligned} \tag{6}$$

where $\mathbf{r}_a$ and $\mathbf{r}_b$ are two camera rays passing through the matched pixel coordinates $\mathbf{u}_a$ and $\mathbf{u}_b$ between two different images a and b, $\{\mathbf{x}_{ak}\}$ and $\{\mathbf{x}_{bk}\}$ are the sampled points along the ray $\mathbf{r}_a$ and $\mathbf{r}_b$, $g_a(\cdot)$ and $g_b(\cdot)$ are the 2D reprojection functions according to the current predicted poses of images a and b, w_{ak} and w_{bk} are the predicted weights for the points on the ray. We use LoFTR (Sun et al. 2021) to generate sparse 2D matches for images within a few frame intervals (typically less than 10). We randomly sample available 2D matches between different frames during training.

Another challenge of pose-free reconstruction is large pose changes in videos, where we observe that the newly appeared surface degrades the previously reconstructed shape. To handle this, we first select B images to jointly train pose and shape networks with fixed iterations, and then add another group of B images and retrain the pose and shape networks with all previous images jointly. During this loop, we periodically reset the shape network to output a unit sphere if the pose network result indicates that the current pose is larger than τ relative to the pose recorded in the most recent resetting. We typically set $\tau = 60^\circ$ in our experiments.

3.5 Refinement with Real Camera

The optimization target $\{(\mathbf{R}_v, \mathbf{t}_v)\}$ of previous sections is based on the virtual camera and not physically-compliant. It is usually not accurately aligned with the original optimization target $\{(\mathbf{R}, \mathbf{t})\}$. To address this problem, we use a PnP solver to transform the predicted pose from the virtual camera system back to the real camera system and refine the results w.r.t. the real camera.

Methods	cracker	sugar	mustard	bleach	meatcan	driller	pitcher	mug	banana	Average
COLMAP	4.08	6.66	4.43	14.11	10.21	11.06	43.38	-	-	13.41
Ye *et al*.	10.21	6.19	**2.61**	**4.18**	3.43	15.15	8.87	-	**3.47**	6.76
Hampali *et al*.	2.91	3.01	4.44	5.63	1.95	5.48	9.21	4.53	4.60	4.64
Ours	**1.71**	**1.84**	3.49	5.38	**1.80**	**3.82**	**2.84**	**2.78**	4.54	**3.14**
UNISURF	3.40	3.49	4.34	3.41	1.54	5.33	4.63	-	3.98	3.76
NeuS	**1.75**	**1.69**	2.34	3.35	**1.17**	**3.13**	3.48	**2.19**	**2.08**	**2.35**
Patten *et al*.	3.54	3.34	3.28	**2.43**	3.26	3.77	4.73	4.22	2.44	3.45

Table 1: Mesh evaluation in $HD_{RMSE} \downarrow$. Our method outperforms most pose-free methods (top group), and is on par with methods trained based on ground truth poses (bottom group).

Methods	cracker	sugar	mustard	bleach	meatcan	driller	pitcher	mug	banana	Average
COLMAP	7.4	7.4	3.5	1.5	0.1	2.8	4.1	2.4	0.0	2.9
Hampali *et al*.	**7.6**	6.8	**5.2**	4.7	**6.8**	6.4	4.6	2.2	**0.6**	4.5
Ours	**7.6**	**7.6**	4.2	**5.3**	3.1	**8.5**	**8.7**	**8.1**	0.3	**5.9**
FoundPose	5.0	3.5	2.1	2.0	0.2	3.0	0.0	1.8	0.0	2.0
MegaPose	7.4	7.4	2.3	0.0	0.0	3.8	0.0	0.1	0.0	2.3

Table 2: Pose evaluation in $AUC_{ATE} \uparrow$. Our method produces more accurate pose result on most objects. Even though methods in the bottom group use ground truth mesh for pose estimation, most of them fail to obtain accurate pose results.

Specifically, we sample a set of 3D points from the reconstructed mesh in previous sections, and re-project them to the virtual image plane with the predicted $(\hat{\mathbf{R}}_v, \hat{\mathbf{t}}_v)$. After transforming the reprojected 2D position with $\mathbf{M}^{-1}$ in Eq. 5, we establish a set of 3D-to-2D correspondences between 3D points and 2D image locations on the raw image, we use RANSAC EPnP (Lepetit, Moreno-Noguer, and Fua 2009) to compute $(\hat{\mathbf{R}}, \hat{\mathbf{t}})$ which is the estimated object pose w.r.t. the real camera. We then conduct a global optimization with all available images starting from $\{(\hat{\mathbf{R}}, \hat{\mathbf{t}})\}$.

Unlike the initialization stage, which models the rays from pixels outside the mask as empty rays, which assumes no 3D point along the ray and is beneficial for fast convergence, but usually results in incomplete meshes for occluded objects, we set those rays as valid, but with zero RGB value in refinement, allowing the shape to benefit from multi-view compensation. We do not use the match loss in refinement, and Fig. 5 shows some results.

3.6 Implementation Details

Networks. Besides the standard SDF MLP and Color MLP used in standard NeuS (Wang et al. 2021b), we use another small Pose MLP to estimate the object pose of the target in every frame. Similar to (Schieber et al. 2023), we map each frame ID to a 256-dim feature vector based on Gaussian Fourier features and then use 3 layers of MLP with GELU activation functions to output the poses.

We use a single ADAM optimizer (Kingma and Ba 2015) for SDP MLP and Color MLP. During training, the learning rate warms up linearly from 0 to 5e-4 during the first 5k iteration and then follows a cosine decay schedule with alpha=0.05. For Pose MLP, we use another ADAM optimizer with a cosine decay schedule of alpha=0.5.

Sampling. We use an unit sphere for the initialization of the SDF network, and define the sampling range (i.e., near and far) within the unit sphere. For each training step, we randomly sample 512 rays from the input image batch. During the optimization with guided virtual camera, we only sample 32 points along each ray for efficiency. We progressively train our model with B consecutive images as a group. For every group, we train the networks with a fixed number of training steps (typically 1K). We sample 20% of the rays from images within previously-converged groups and 80% from the images within the newly added group.

In the phase of refinement with the real camera, we use importance-based hierarchy sampling strategies (Wang et al. 2021b) to uniformly sample 64 points and then sample another 64 points based on the current predicted SDF values.

We train the networks for 150K training steps for refinement. On a typical NVIDIA V100 GPU, the training of a 100-frame sequence takes about 3 hours for initialization and 7 hours for refinement.

4 Experiments

We evaluate our method systematically in this section.

Datasets. We first evaluate our method on the standard HO3D dataset (Hampali et al. 2020), which includes video captures of daily objects with a fixed camera. Since most objects in HO3D are manipulated by one hand, and there exists a fixed relative pose between the hand and the object in most sequences, we collect a more general dataset of free-moving objects with a head-mounted device with egocentric views, where the objects are manipulated by both hands with a free manipulation style.

Metrics. For object reconstruction, we first align the predicted mesh to the GT mesh based on scale-aware

ICP (Hampali et al. 2023), then calculate the root mean square of the Hausdorff distance (mm) (Hampali et al. 2023; Patten et al. 2021) between the predicted mesh and the GT mesh (HD_{RMSE}). For object pose estimation, we first align predicted poses to GT poses based on similarity transforms (Hampali et al. 2023; Bian et al. 2023; Fu et al. 2024), and then compute the area under curve (AUC) with a threshold of 10cm in Absolute Trajectory Error (ATE) (AUC_{ATE}) (Hampali et al. 2023). We also report the pose accuracy in Relative Pose Error (RPE) (Bian et al. 2023; Fu et al. 2024), including RPE_r (degree) and RPE_t (cm), corresponding to the rotation error and the translation error, respectively.

4.1 Evaluation on HO3D

We report results on the 9 sequences of HO3D as in (Hampali et al. 2023; Ye, Gupta, and Tulsiani 2022), and compare our method with pose-free methods including COLMAP (Schönberger and Frahm 2016), Ye *et al*. (Ye, Gupta, and Tulsiani 2022), and Hampali *et al*. (Hampali et al. 2023), where Ye *et al*. relies on prior hand pose information. We also compare with FoundPose (Örnek et al. 2024) and MegaPose (Labbé et al. 2022) that predict object pose based on GT meshes. To further validate our method, we compare our reconstructed mesh with the results of UNISURF (Oechsle, Peng, and Geiger 2021) and NeuS (Wang et al. 2021b) which are trained with ground truth poses, and also another method Patten *et al*. (Patten et al. 2021) that rely on depth images.

We summarize the reconstruction results in Table 1. COLMAP can not produce accurate results on most objects, mainly caused by the lacking of enough textures, especially for objects like bleach and pitcher in the table. With prior hand pose information, Ye's method produces better results than COLMAP. With carefully selected multiple easy segments, Hampali's method outperforms Ye's method. Nevertheless, our method produces better results than most of them, and is even on par with methods relying on ground truth poses (UNISURF and NeuS) or depth images (Patten's method). Fig. 6 shows some visualization results of the reconstruction meshes.

Table 2 shows the evaluation of pose results. Our method outperforms COLMAP and Hampali's method significantly. On the other hand, FoundPose and MegaPose use the render-and-compare strategy to predict the pose based on ground truth mesh, but most of them fail to obtain accurate pose results. We found that their predicted pose is reasonably well for most frames in the sequence. However, there are a significant number of pose outliers, especially for textureless and symmetry objects under some views, which produces much worse numbers than our method in AUC_{ATE}.

4.2 Ablation Study

We evaluate the design of our method systematically in this section. We report the results averaging across all the 9 sequences in HO3D, if not explicitly mentioned.

Effectiveness of virtual camera. We study the effect of the proposed virtual camera in Table 3. In principle, one

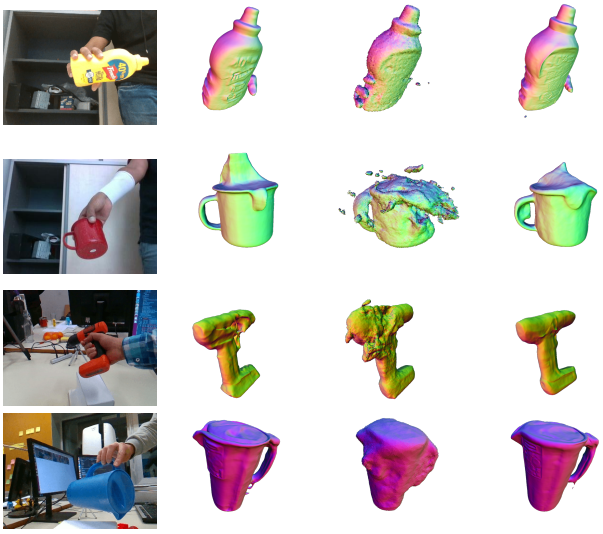


Figure 6: Our method produces significantly better results than Hampali *et al*.'s method, and is even on par with NeuS that is trained with ground truth poses.

Strategy	$AUC_{ATE}\uparrow$	Pose $RPE_t\downarrow$	$RPE_r\downarrow$	Mesh $HD_{RMSE}\downarrow$
RC	4.33	4.14	7.19	4.15
VC	5.42	1.94	2.65	3.90
VC + RC	**5.93**	**1.57**	**2.20**	3.14
Oracle	-	-	-	**2.25**

Table 3: Effect of virtual cameras. Progressively training w.r.t. the real camera ("RC") typically fails. The proposed virtual camera system ("VC") reduces the search space and improves the results significantly. The global refinement w.r.t. the real camera ("+RC") improves the performance further, producing meshes that most closely resemble "Oracle" which is trained using ground truth pose.

Settings	$AUC_{ATE}\uparrow$	Pose $RPE_t\downarrow$	$RPE_r\downarrow$	Mesh $HD_{RMSE}\downarrow$
w/o match	5.21	1.87	2.38	4.10
w/o reset	5.58	1.64	2.26	3.28
w/o 4D	5.68	2.26	3.49	3.27
Full	**5.93**	**1.57**	**2.20**	**3.14**

Table 4: Ablation study. We evaluate different components of our method, including match loss ("match"), periodical reset of shape networks ("reset"), and reducing the 6 degrees of freedom of poses to 4 ("4D").

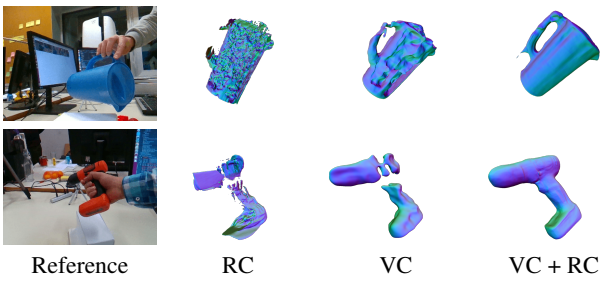


Figure 7: Effect of virtual camera. Progressively training w.r.t. the real camera ("RC") is challenging and typically fails. The proposed virtual camera system ("VC") reduces the search space of optimization and improves the results significantly after refinement w.r.t. the real camera ("+RC").

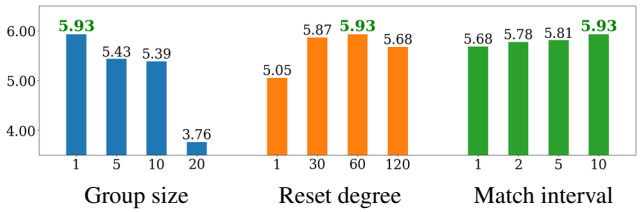


Figure 8: Ablation study of hyperparameters in AUC_{ATE} ↑.

could leverage the temporal consistency between consecutive frames simply by introducing the progressive training, similar to (Bian et al. 2023; Fu et al. 2024). We compare this strategy with our method. For a fair comparison, we use the same network and the same segmentation mask and 2D matches as those used in our method. The major problem of this strategy is that the pose and shape are optimized w.r.t. the raw camera, which is challenging due to the target's free movement and the wide range it covers in front of the camera. We denote this method as "RC" in the table. By contrast, the proposed virtual camera reduces the search space and improves the results significantly ("VC"). The result improves further with the refinement w.r.t the real camera ("+RC"). Fig. 7 shows some visualization results.

Ablation study of different components. We study the effect of different components of our method in Table 4. Without the match loss or the periodic reset of the shape network, the performance has a significant drop (denoted as "w/o match" and "w/o reset" respectively). On the other hand, using the standard 6D pose representation (denoted as "w/o 4D") also suffers in performance. We use the same refinement procedure for all the results in this table.

Ablation study of different hyperparameters. We show ablation results of our method with different hyperparameters in Fig. 8, including the number of frames B that is processed as a group during progressing training, the degree threshold τ for periodic reset of the shape network, and the the maximum frame internal n for 2D matching across consecutive images. As we can see, B=1 gives the best results, and as the number of images in each group increases, the performance deteriorates, which we believe is caused by the difficulties introduced with the increased data varieties. On the other hand, τ=60° and n=10 produce the best results in most of our experiments.

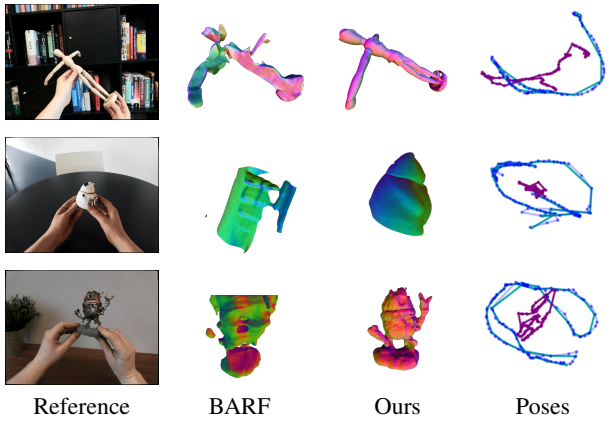


Figure 9: Results on egocentric sequences.

4.3 Results on Egocentric Sequences

To verify the generalization ability of our method, we capture some sequences with a head-mounted AR device with egocentric views, where the camera is naturally moving with the user's head and the target is freely manipulated by the user. We illustrate the results in Fig. 9. To get ground truth poses for evaluation, we manually annotate pose every 5 frames for each sequence with depth images as guidance. While, note that our method relies only on RGB images and does not use depth images in both training and inference. The result shows that our method generalizes well to this real setting, and produces accurate results for free-moving daily objects.

Limitation. Although our method produces accurate pose and mesh results in most cases, it cannot handle scenarios where some parts of the object are occluded for a long time during capture. On the other hand, our method still cannot produce accurate results for tiny texture-less objects due to the lacking of enough features. Addressing this will be one of our future work.

5 Conclusion

We have showed that it is possible to jointly optimize the reconstruction and pose estimation for free-moving objects without relying on any prior information, or any segmenting procedure from a monocular RGB video. It relies on the intuition that, using estimated 2D object masks, one can reformulate the joint optimization problem w.r.t. a virtual camera pointing to the object center, which simplifies the trajectory and reduces the search space of optimization significantly. Although the pose and shape are not physically-compliant in the virtual camera system, our experiments have demonstrated that optimizing w.r.t. the virtual camera yields robust initialization results and produces accurate final results after a refinement w.r.t. the real camera.

References

Agarwal, S.; Furukawa, Y.; Snavely, N.; Simon, I.; Curless, B.; Seitz, S. M.; and Szeliski, R. 2011. Building Rome in a day. *Commun. ACM*.

Bian, W.; Wang, Z.; Li, K.; and Bian, J. 2023. NoPe-NeRF: Optimising Neural Radiance Field with No Pose Prior. *IEEE/CVF Computer Vision and Pattern Recognition Conference (CVPR)*.

Billinghurst, M. 2021. Grand Challenges for Augmented Reality. *Frontiers in Virtual Reality*.

Chen, D.; Li, J.; Wang, Z.; and Xu, K. 2020. Learning Canonical Shape Space for Category-Level 6D Object Pose and Size Estimation. *IEEE/CVF Computer Vision and Pattern Recognition Conference (CVPR)*.

Chen, J.; Yan, M.; Zhang, J.; Xu, Y.; Li, X.; Weng, Y.; Yi, L.; Song, S.; and Wang, H. 2023. Tracking and Reconstructing Hand Object Interactions from Point Cloud Sequences in the Wild. *The AAAI Conference on Artificial Intelligence (AAAI)*.

Cheng, H. K.; and Schwing, A. G. 2022. XMem: Long-Term Video Object Segmentation with an Atkinson-Shiffrin Memory Model. *European Conference on Computer Vision (ECCV)*.

Cheng, Z.; Li, H.; Asano, Y.; Zheng, Y.; and Sato, I. 2021. Multi-view 3D Reconstruction of a Texture-less Smooth Surface of Unknown Generic Reflectance. *IEEE/CVF Computer Vision and Pattern Recognition Conference (CVPR)*.

Fan, Z.; Parelli, M.; Kadoglou, M. E.; Kocabas, M.; Chen, X.; Black, M. J.; and Hilliges, O. 2024. HOLD: Category-agnostic 3D Reconstruction of Interacting Hands and Objects from Video. *IEEE/CVF Computer Vision and Pattern Recognition Conference (CVPR)*.

Fu, Y.; Liu, S.; Kulkarni, A.; Kautz, J.; Efros, A. A.; and Wang, X. 2024. COLMAP-Free 3D Gaussian Splatting. *IEEE/CVF Computer Vision and Pattern Recognition Conference (CVPR)*.

Hampali, S.; Hodan, T.; Tran, L.; Ma, L.; Keskin, C.; and Lepetit, V. 2023. In-Hand 3D Object Scanning from an RGB Sequence. *IEEE/CVF Computer Vision and Pattern Recognition Conference (CVPR)*.

Hampali, S.; Rad, M.; Oberweger, M.; and Lepetit, V. 2020. HOnnotate: A Method for 3D Annotation of Hand and Object Poses. *IEEE/CVF Computer Vision and Pattern Recognition Conference (CVPR)*.

Hodan, T.; Michel, F.; Brachmann, E.; Kehl, W.; GlentBuch, A.; Kraft, D.; Drost, B.; Vidal, J.; Ihrke, S.; Zabulis, X.; Sahin, C.; Manhardt, F.; Tombari, F.; Kim, T.-K.; Matas, J.; and Rother, C. 2018. BOP: Benchmark for 6D Object Pose Estimation. *European Conference on Computer Vision (ECCV)*.

Hu, Y.; Fua, P.; and Salzmann, M. 2022. Perspective Flow Aggregation for Data-Limited 6D Object Pose Estimation. *European Conference on Computer Vision (ECCV)*.

Hu, Y.; Speierer, S.; Jakob, W.; Fua, P.; and Salzmann, M. 2021. Wide-Depth-Range 6D Object Pose Estimation in Space. *IEEE/CVF Computer Vision and Pattern Recognition Conference (CVPR)*.

Huang, D.; Ji, X.; He, X.; Sun, J.; He, T.; Shuai, Q.; Ouyang, W.; and Zhou, X. 2022. Reconstructing Hand-Held Objects from Monocular Video. *SIGGRAPH Asia*.

Jeong, Y.; Ahn, S.; Choy, C.; Anandkumar, A.; Cho, M.; and Park, J. 2021. Self-Calibrating Neural Radiance Fields. *IEEE/CVF International Conference on Computer Vision (ICCV)*.

Jiang, S.; Ye, Q.; Xie, R.; Huo, Y.; Li, X.; Zhou, Y.; and Chen, J. 2024. In-Hand 3D Object Reconstruction from a Monocular RGB Video. *The AAAI Conference on Artificial Intelligence (AAAI)*.

Kajiya, J. T.; and Von Herzen, B. P. 1984. RAY TRACING VOLUME DENSITIES. *ACM SIGGRAPH*.

Kingma, D. P.; and Ba, J. 2015. Adam: A Method for Stochastic Optimization. *The International Conference on Learning Representations (ICLR)*.

Kirillov, A.; Mintun, E.; Ravi, N.; Mao, H.; Rolland, C.; Gustafson, L.; Xiao, T.; Whitehead, S.; Berg, A. C.; Lo, W.-Y.; Dollár, P.; and Girshick, R. B. 2023. Segment Anything. *IEEE/CVF International Conference on Computer Vision (ICCV)*.

Labbé, Y.; Manuelli, L.; Mousavian, A.; Tyree, S.; Birchfield, S.; Tremblay, J.; Carpentier, J.; Aubry, M.; Fox, D.; and Sivic, J. 2022. MegaPose: 6D Pose Estimation of Novel Objects via Render & Compare. *Proceedings of the 6th Conference on Robot Learning (CoRL)*.

Lepetit, V.; Moreno-Noguer, F.; and Fua, P. 2009. EPnP: An Accurate O(n) Solution to the PnP Problem. *International Journal of Computer Vision (IJCV)*.

Lin, C.; Ma, W.; Torralba, A.; and Lucey, S. 2021. BARF: Bundle-Adjusting Neural Radiance Fields. *IEEE/CVF International Conference on Computer Vision (ICCV)*.

Lorensen, W. E.; and Cline, H. E. 1987. Marching Cubes: A High Resolution 3D Surface Construction Algorithm. *SIGGRAPH*.

Mildenhall, B.; andMatthew Tancik, P. P. S.; Barron, J. T.; Ramamoorthi, R.; and Ng, R. 2020. NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis. *European Conference on Computer Vision (ECCV)*.

Mohr, R.; Quan, L.; and Veillon, F. 1995. Relative 3D Reconstruction Using Multiple Uncalibrated Images. *The International Journal of Robotics Research*.

Moreno-Noguer, F.; Lepetit, V.; and Fua, P. 2007. Accurate Non-Iterative O (n) Solution to the P n P Problem. *IEEE/CVF International Conference on Computer Vision (ICCV)*.

Müller, T.; Evans, A.; Schied, C.; and Keller, A. 2022. Instant Neural Graphics Primitives with a Multiresolution Hash Encoding. *ACM TOG*.

Oechsle, M.; Peng, S.; and Geiger, A. 2021. UNISURF: Unifying Neural Implicit Surfaces and Radiance Fields for Multi-View Reconstruction. *IEEE/CVF International Conference on Computer Vision (ICCV)*.

Örnek, E. P.; Labbé, Y.; Tekin, B.; Ma, L.; Keskin, C.; Forster, C.; and Hodaň, T. 2024. FoundPose: Unseen Object Pose Estimation with Foundation Features. *European Conference on Computer Vision (ECCV)*.

Patten, T.; Park, K.; Leitner, M.; Wolfram, K.; and Vincze, M. 2021. Object Learning for 6D Pose Estimation and Grasping from RGB-D Videos of In-hand Manipulation. *The IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*.

Peng, W.; Yan, J.; Wen, H.; and Sun, Y. 2022. Self-Supervised Category-Level 6D Object Pose Estimation with Deep Implicit Shape Representation. *The AAAI Conference on Artificial Intelligence (AAAI)*.

Pollefeys, M.; Van Gool, L.; Vergauwen, M.; Verbiest, F.; Cornelis, K.; Tops, J.; and Koch, R. 2004. Visual Modeling with a Hand-Held Camera. *International Journal of Computer Vision (IJCV)*.

Pumarola, A.; Corona, E.; Pons-Moll, G.; and Moreno-Noguer, F. 2021. D-NeRF: Neural Radiance Fields for Dynamic Scenes. *IEEE/CVF Computer Vision and Pattern Recognition Conference (CVPR)*.

Rosinol, A.; Leonard, J. J.; and Carlone, L. 2022. NeRF-SLAM: Real-Time Dense Monocular SLAM with Neural Radiance Fields. *arXiv*.

Rozumnyi, D.; Matas, J.; Pollefeys, M.; Ferrari, V.; and Oswald, M. R. 2023. Tracking by 3D Model Estimation of Unknown Objects in Videos. *IEEE/CVF International Conference on Computer Vision (ICCV)*.

Rusinkiewicz, S.; Hall-Holt, O.; and Levoy, M. 2002. Real-Time 3D Model Acquisition. *ACM TOG*.

Rünz, M.; Li, K.; Tang, M.; Ma, L.; Kong, C.; Schmidt, T.; Reid, I.; Agapito, L.; Straub, J.; Lovegrove, S.; and Newcombe, R. 2020. FroDO: From Detections to 3D Objects. *IEEE/CVF Computer Vision and Pattern Recognition Conference (CVPR)*.

Sabae, M. S.; Baraka, H. A.; and Hadhoud, M. M. 2023. NoPose-NeuS: Jointly Optimizing Camera Poses with Neural Implicit Surfaces for Multi-view Reconstruction. *arXiv*.

Schieber, H.; Deuser, F.; Egger, B.; Oswald, N.; and Roth, D. 2023. NeRFtrinsic Four: An End-To-End Trainable NeRF Jointly Optimizing Diverse Intrinsic and Extrinsic Camera Parameters. *arXiv*.

Schönberger, J. L.; and Frahm, J. 2016. Structure-from-Motion Revisited. *IEEE/CVF Computer Vision and Pattern Recognition Conference (CVPR)*.

Snavely, N. 2011. Scene Reconstruction and Visualization from Internet Photo Collections: A Survey. *IPSJ Trans. Comput. Vis. Appl.*

Snavely, N.; Seitz, S. M.; and Szeliski, R. 2006. Photo Tourism: Exploring Photo Collections in 3D. *ACM TOG*.

Su, Y.; Saleh, M.; Fetzer, T.; Rambach, J.; Navab, N.; Busam, B.; Stricker, D.; and Tombari, F. 2022. ZebraPose: Coarse to Fine Surface Encoding for 6DoF Object Pose Estimation. *IEEE/CVF Computer Vision and Pattern Recognition Conference (CVPR)*.

Sun, J.; Shen, Z.; Wang, Y.; Bao, H.; and Zhou, X. 2021. LoFTR: Detector-Free Local Feature Matching with Transformers. *IEEE/CVF Computer Vision and Pattern Recognition Conference (CVPR)*.

Sweeney, C. 2016. Theia Multiview Geometry Library: Tutorial & Reference. http://theia-sfm.org.

Thalhammer, S.; Bauer, D.; Hönig, P.; Weibel, J.-B.; García-Rodríguez, J.; and Vincze, M. 2023. Challenges for Monocular 6D Object Pose Estimation in Robotics. *arXiv*.

Tian, M.; Ang, M. H.; and Lee, G. H. 2020. Shape Prior Deformation for Categorical 6D Object Pose and Size Estimation. *European Conference on Computer Vision (ECCV)*.

Tzionas, D.; and Gall, J. 2015. 3D Object Reconstruction from Hand-Object Interactions. *IEEE/CVF International Conference on Computer Vision (ICCV)*.

Wang, G.; Manhardt, F.; Tombari, F.; and Ji, X. 2021a. GDR-Net: Geometry-Guided Direct Regression Network for Monocular 6D Object Pose Estimation. *IEEE/CVF Computer Vision and Pattern Recognition Conference (CVPR)*.

Wang, P.; Liu, L.; Liu, Y.; Theobalt, C.; Komura, T.; and Wang, W. 2021b. NeuS: Learning Neural Implicit Surfaces by Volume Rendering for Multi-view Reconstruction. *Conference and Workshop on Neural Information Processing Systems (NeurIPS)*.

Wang, Y.; Han, Q.; Habermann, M.; Daniilidis, K.; Theobalt, C.; and Liu, L. 2023. NeuS2: Fast Learning of Neural Implicit Surfaces for Multi-view Reconstruction. *IEEE/CVF International Conference on Computer Vision (ICCV)*.

Weise, T.; Leibe, B.; and Van Gool, L. 2008. Accurate and Robust Registration for In-hand Modeling. *IEEE/CVF Computer Vision and Pattern Recognition Conference (CVPR)*.

Weise, T.; Wismer, T.; Leibe, B.; and Van Gool, L. 2009. In-hand Scanning with Online Loop Closure. *IEEE/CVF International Conference on Computer Vision Workshops (ICCVW)*.

Weng, Y.; Wang, H.; Zhou, Q.; Qin, Y.; Duan, Y.; Fan, Q.; Chen, B.; Su, H.; and Guibas, L. J. 2021. CAPTRA: CAtegory-level Pose Tracking for Rigid and Articulated Objects from Point Clouds. *IEEE/CVF International Conference on Computer Vision (ICCV)*.

Yariv, L.; Gu, J.; Kasten, Y.; and Lipman, Y. 2021. Volume Rendering of Neural Implicit Surfaces. *Conference and Workshop on Neural Information Processing Systems (NeurIPS)*.

Yariv, L.; Kasten, Y.; Moran, D.; Galun, M.; Atzmon, M.; Ronen, B.; and Lipman, Y. 2020. Multiview Neural Surface Reconstruction by Disentangling Geometry and Appearance. *Conference and Workshop on Neural Information Processing Systems (NeurIPS)*.

Ye, Y.; Gupta, A.; and Tulsiani, S. 2022. What's in your hands? 3D Reconstruction of Generic Objects in Hands. *IEEE/CVF Computer Vision and Pattern Recognition Conference (CVPR)*.

Yu, S.; Zhai, D.; and Xia, Y. 2024. CatFormer: Category-Level 6D Object Pose Estimation with Transformer. *The AAAI Conference on Artificial Intelligence (AAAI)*.

Zou, Z.; Cheng, W.; Cao, Y.; Huang, S.; Shan, Y.; and Zhang, S. 2024. Sparse3D: Distilling Multiview-Consistent Diffusion for Object Reconstruction from Sparse Views. *The AAAI Conference on Artificial Intelligence (AAAI)*.