AlignSDF: Pose-Aligned Signed Distance Fields for Hand-Object Reconstruction

Zerui Chen, Yana Hasson, Cordelia Schmid, Ivan Laptev

AlignSDF: Pose-Aligned Signed Distance Fields for Hand-Object Reconstruction

Zerui Chen[1], Yana Hasson[1,2], Cordelia Schmid[1], and Ivan Laptev[1]

[1] Inria, École normale supérieure, CNRS, PSL Research Univ., 75005 Paris, France
[2] Now at Deepmind
firstname.lastname@inria.fr
https://zerchen.github.io/projects/alignsdf.html

Abstract. Recent work achieved impressive progress towards joint reconstruction of hands and manipulated objects from monocular color images. Existing methods focus on two alternative representations in terms of either parametric meshes or signed distance fields (SDFs). On one side, parametric models can benefit from prior knowledge at the cost of limited shape deformations and mesh resolutions. Mesh models, hence, may fail to precisely reconstruct details such as contact surfaces of hands and objects. SDF-based methods, on the other side, can represent arbitrary details but are lacking explicit priors. In this work we aim to improve SDF models using priors provided by parametric representations. In particular, we propose a joint learning framework that disentangles the pose and the shape. We obtain hand and object poses from parametric models and use them to align SDFs in 3D space. We show that such aligned SDFs better focus on reconstructing shape details and improve reconstruction accuracy both for hands and objects. We evaluate our method and demonstrate significant improvements over the state of the art on the challenging ObMan and DexYCB benchmarks.

Keywords: Hand-object reconstruction, Parametric mesh models, Signed distance fields (SDFs)

1 Introduction

Reconstruction of hands and objects from visual data holds a promise to unlock widespread applications in virtual reality, robotic manipulation and human-computer interaction. With the advent of deep learning, we have witnessed a large progress towards 3D reconstruction of hands [4, 5, 24, 39, 57, 74] and objects [12, 16, 47, 66]. Joint reconstruction of hands and manipulated objects, as well as detailed modeling of hand-object interactions, however, remains less explored and poses additional challenges.

Some of the previous works explore 3D cues and perform reconstruction from multi-view images [7], depth maps [1, 61, 73] or point clouds [9]. Here, we focus on a more challenging but also more practical setup and reconstruct hands and objects jointly from monocular RGB images. Existing methods in this setting can be generally classified as the ones using parametric mesh models [20, 21, 49, 54, 71] and methods based on implicit representations [11, 26, 36, 45].

Methods from the first category [20, 21, 71] often build on MANO [54], a popular parametric hand model, see Figure 1(a). Since MANO is derived from 3D scans of

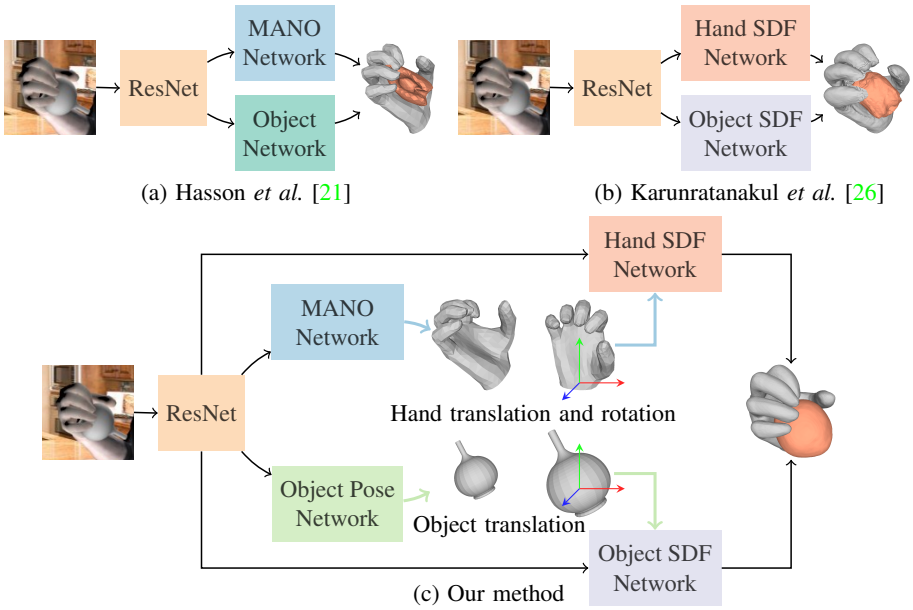


Fig. 1: Previous work on hand-object reconstruction use either (a) parametric shape models or (b) implicit 3D representations. Our proposed method (c) extends SDFs with prior knowledge on hand and object poses obtained via parametric models and can produce detailed meshes for hands and manipulated objects from monocular RGB images.

real human hands and encodes strong prior shape knowledge, such methods typically provide anthropomorphically valid hand meshes. However, the resolution of parametric meshes is limited, making them hard to recover detailed interactions. Also, reconstructing 3D objects remains a big challenge. Hasson *et al.* [21] propose to use AtlasNet [16] to reconstruct 3D objects. However, their method can only reconstruct simple objects, and the reconstruction accuracy remains limited. To improve reconstruction, several methods [20, 62, 71] make a restricting assumption that the ground-truth 3D object model is available at test time and only predict the 6D pose of the object.

Recently, neural implicit representations have shown promising results for object reconstruction [45]. Following this direction, Karunratanakul *et al.* [26] propose to represent hands and objects in a unified signed distance field (SDF) and show the potential to model hand-object interactions, see Figure 1(b). We here adopt SDF and argue that such implicit representations may benefit from explicit prior knowledge about the pose of hands and objects.

For more accurate reconstruction of hands and manipulated objects, we attempt to combine the advantages of the parametric models and SDFs. Along this direction, previous works [10, 14, 25, 56] attempt to leverage parametric models to learn SDFs from 3D poses or raw scans. In our work, we address a different and more challenging setup of reconstructing hands and objects from monocular RGB images. We hence propose a new pose-normalized SDF framework suited for our task.

Scene geometry depends both on the shape and the global pose of underlying objects. While the pose generally affects all object points with low-parametric transformations (e.g., translation and rotation), it is a common practice to separate the pose and the shape parameters of the model [13, 54]. We, hence, propose to disentangle the learning of pose and shape for both hands and objects. As shown in Figure 1(c), for the hand, we first estimate its MANO parameters and then learn hand SDF in a canonical frame normalized with respect to the rotation and trsnslation of the hand wrist. Similarly, for objects, we estimate their translation and learn object SDF in a translation-normalized canonical frame. By normalizing out the pose, we simplify the task of SDF learning which can focus on estimating the shape disregarding the global rotation and translation transformations. In our framework, the MANO network and the object pose network are responsible for solving the pose, and SDF networks focus on learning the geometry of the hand and the object under their canonical poses.

To validate the effectiveness of our approach, we conduct extensive experiments on two challenging benchmarks: ObMan [21] and DexYCB [6]. ObMan is a synthetic dataset and contains a wide range of objects and grasp types. DexYCB is currently the largest real dataset for capturing hands and manipulated objects. We experimentally demonstrate that our approach outperforms state-of-the-art methods by a significant margin on both benchmarks. Our contributions can be summarized as follows:

• We propose to combine the advantages of parametric mesh models and SDFs and present a joint learning framework for 3D reconstruction of hands and objects.

• To effectively incorporate prior knowledge into SDFs learning, we propose to disentangle the pose learning from the shape learning for this task. Within our framework, we employ parametric models to estimate poses for the hand and the object and employ SDF networks to learn hand and object shapes in pose-normalized coordinate frames.

• We show the advantage of our method by conducting comprehensive ablation experiments on ObMan. Our method produces more detailed joint reconstruction results and achieves state-of-the-art accuracy on the ObMan and DexYCB benchmarks.

2 Related Work

Our work focuses on joint reconstruction of hands and manipulated objects from monocular RGB images. In this section, we first review recent methods for object shape modeling and 3D hand reconstruction. Then, we focus on hand-and-object interaction modeling from a single color image.

3D object modeling. Modeling the pose and shape of 3D objects from monocular images is one of the longest standing objectives of computer vision [42, 52]. Recent methods train deep neural network models to compute the object shape [11, 16, 36, 55, 70] and pose [31, 32, 34, 68] directly from image pixels. Learned object shape reconstruction from single view images has initially focused on point-cloud [48], mesh [16, 66] and voxel [12, 51] representations. In recent years, deep implicit representations [11, 36, 45] have gained popularity. Unlike other commonly used representations, implicit functions can theoretically model surfaces at unlimited resolution, which makes them an ideal choice to model detailed interactions. We propose to leverage the flexibility of implicit functions to reconstruct hands and arbitrary unknown objects. By conditioning

the signed distance function (SDF) on predicted poses, we can leverage strong shape priors from available models. Recent work [59] also reveals that it is effective to encode structured information to improve the quality of NeRF [37] for articulated bodies.

3D hand reconstruction. The topic of 3D hand reconstruction has attracted wide attention since the 90s [23, 50]. In the deep learning era, we have witnessed significant progress in hand reconstruction from color images. Most works focus on predicting 3D positions of sparse keypoints [24, 38, 40, 60, 69, 75]. These methods can achieve high accuracy by predicting each hand joint locations independently. However sparse representations of the hand are insufficient to reason precisely about hand-object interactions, which requires millimeter level accuracy. To address this limitation, several recent works model the dense hand surface [2, 4, 5, 8, 29, 30, 41, 44, 65, 74]. A popular line of work reconstructs the hand surface by estimating the parameters of MANO [54], a deformable hand mesh model. These methods can produce anthropomorphically plausible hand meshes using the strong hand prior captured by the parametric model. Such methods either learn to directly regress hand mesh parameters from RGB images [2, 4, 8, 74] or fit them to a set of constraints as a post-processing step [41, 44, 65]. Unlike previous methods, we condition the hand implicit representation on MANO parameters and produce hand reconstructions of improved visual quality.

3D hand-object reconstruction. Joint reconstruction of hands and objects from monocular RGB images is a very challenging task given the partial visibility and strong mutual occlusions. Methods often rely on multi-view images [3, 19, 43, 67] or additional depth information [17, 18, 58, 63, 64] to solve this problem. Recent learning-based methods focus on reconstructing hands and objects directly from single-view RGB images. To simplify the reconstruction task, several methods [15, 20, 62, 71] make a strong assumption that the ground-truth object model is known at test-time and predict its 6D pose. Some methods propose to model hand interactions with unseen objects at test time [21, 28, 53]. Most related to our approach, Hasson *et al.* [21] propose a two-branch network to reconstruct the hand and an unknown manipulated object. The object branch uses AtlasNet [16] to reconstruct the object mesh and estimate its position relative to the hand. Their method can only reconstruct simple objects which can be obtained by deforming a sphere. In contrast, SDF allows us to model arbitrary object shapes.

In order to improve the quality of hand-object reconstructions, [21] introduce heuristic interaction penalties at train time, Yang *et al.* [71] model each hand-object contact as a spring-mass system and refine the reconstruction result by an optimization process. Recent work [33] also applies an online data augmentation strategy to boost the joint reconstruction accuracy. Though these methods based on parametric mesh models can achieve relatively robust reconstruction results, the modeling accuracy is limited by the underlying parametric mesh. Closest to our approach, Karunratanakul *et al.* [26] propose to model the hand, the object and their contact areas using deep signed distance functions. Their method can reconstruct hand and object meshes at a high resolution and capture detailed interactions. However, their method is model-free and does not benefit from any prior knowledge about hands or objects. A concurrent work [72] uses an off-the-shelf hand pose estimator and leverages hand poses to improve hand-held object shapes, which operates in a less-challenging setting than ours. Different from previous works, our method brings together the advantages of both parametric models

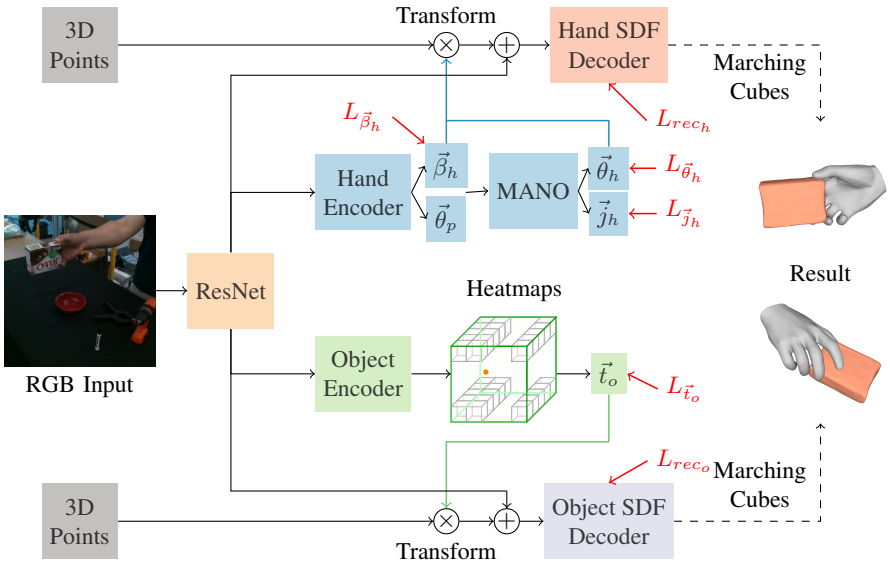


Fig. 2: Our method can reconstruct detailed hand meshes and object meshes from monocular RGB images. Two gray blocks of 3D points indicate the same set of 3D query points. The red arrows denote different loss functions applied during training. The dashed arrows denote Marching Cubes algorithm [35] used at test time.

and deep implicit functions. By embedding prior knowledge into SDFs learning, our method can produce more robust and detailed reconstruction results.

3 Method

As illustrated in Figure 2, our method is designed to reconstruct the hand and object meshes from a single RGB image. Our model can be generally split into two parts: the hand part and the object part. The hand part estimates MANO parameters and uses them to transform 3D points to the hand canonical coordinate frame. Then, the hand SDF decoder predicts the signed distance for each input 3D point and uses the Marching Cubes algorithm [35] to reconstruct the hand mesh at test time. Similarly, the object part estimates the object translation relative to the hand wrist and uses it to transform the same set of 3D points. The object SDF decoder takes the transformed 3D points as input and reconstructs the object mesh. In the following, we describe the three main components of our model: hand pose estimation in Section 3.1, object pose estimation in Section 3.2, and hand and object shape reconstruction in Section 3.3.

3.1 Hand pose estimation

To embed more prior knowledge about human hands into our model, following previous works [20, 21, 71], we employ a parametric hand mesh model, MANO [54], to capture the kinematics for the human hand. MANO is a statistical model, which could map pose ($\vec{\theta}_p$) and shape ($\vec{\beta}_h$) parameters to a hand mesh. To estimate hand poses, we first feed features extracted from ResNet-18 [22] to the hand encoder network. The hand encoder network consists of fully connected layers and regresses $\vec{\theta}_p$ and $\vec{\beta}_h$. Then, we integrate MANO as a differentiable layer into our model and use it to predict the hand vertices ($\vec{v}_h$), the hand joints ($\vec{j}_h$) and hand poses ($\vec{\theta}_h$).

We define the supervision on the joint locations ($L_{\vec{j}_h}$), the shape parameters ($L_{\vec{\beta}_h}$) and the predicted hand poses ($L_{\vec{\theta}_h}$). To compute $L_{\vec{j}_h}$, we apply L2 loss between predicted hand joints and the ground truth. However, using $L_{\vec{j}_h}$ alone can result in extreme mesh deformations [21]. Therefore, we use another two regularization terms: $L_{\vec{\beta}_h}$ and $L_{\vec{\theta}_h}$. The shape regularization term ($L_{\vec{\beta}_h}$) constrains that the predicted hand shape ($\vec{\beta}_h \in \mathbb{R}^{10}$) is close to the mean shape in the MANO training set. The predicted hand poses ($\vec{\theta}_h \in \mathbb{R}^{48}$) consist of axis-angle rotation representations for sixteen joints, including one global rotation for the wrist joint and fifteen rotations for the other local joints. The pose regularization term ($L_{\vec{\theta}_h}$) constrains local joint rotations to be close to the mean pose in the MANO training set. We also apply L2 loss for the two regularization terms. For the task of hand pose estimation, the overall loss L_{hand} is the summation of all $L_{\vec{j}_h}$, $L_{\vec{\beta}_h}$ and $L_{\vec{\theta}_h}$ terms:

$$L_{hand} = \lambda_{\vec{j}_h} L_{\vec{j}_h} + \lambda_{\vec{\beta}_h} L_{\vec{\beta}_h} + \lambda_{\vec{\theta}_h} L_{\vec{\theta}_h}, \tag{1}$$

where we set $\lambda_{\vec{j}_h}$, $\lambda_{\vec{\beta}_h}$ and $\lambda_{\vec{\theta}_h}$ to 5×10^{-1}, 5×10^{-7} and 5×10^{-5}, respectively.

3.2 Object pose estimation

In our method, we set the origin of our coordinate system as the wrist joint defined in MANO. To solve the task of object pose estimation, we usually need to predict the object rotation and its translation. However, estimating the 3D rotation for unknown objects is a challenging and ambiguous task, especially for symmetric objects. Therefore, we here only predict the 3D object translation relative to the hand wrist. To estimate the relative 3D translation ($\vec{t}_o$), we employ volumetric heatmaps [38, 46] to predict per voxel likelihood for the object centroid and use a soft-argmax operator [60] to extract the 3D coordinate from heatmaps. Then, we convert the 3D coordinate into our wrist-relative coordinate system using camera intrinsics and the wrist location.

During training, we optimize network parameters by minimizing the L2 loss between the estimated 3D object translations $\vec{t}_o$ and corresponding ground truth. For the task of object pose estimation, the resulting loss L_{obj} is the summation of $L_{\vec{t}_o}$:

$$L_{obj} = \lambda_{\vec{t}_o} L_{\vec{t}_o}, \tag{2}$$

where we empirically set $\lambda_{\vec{t}_o}$ to 5×10^{-1}.

3.3 Hand and object shape reconstruction

Following previous works [26, 45], we use neural networks to approximate signed distance functions for the hand and the object. For any input 3D point $\vec{x}$, we employ the hand SDF decoder and the object SDF decoder to predict its signed distance to the hand surface and the object surface, respectively. However, it is very challenging to directly learn neural implicit representations for this task, because SDF networks have to handle a wide range of objects and different types of grasps. As a result, Grasping Field [26] cannot achieve satisfactory results in producing detailed hand-and-object interactions.

To reduce the difficulty for this task, our method makes an attempt to disentangle the shape learning and the pose learning, which could help liberate the power of SDF networks. By estimating the hand pose, we could obtain the global rotation ($\vec{\theta}_{hr}$) and its rotation center ($\vec{t}_h$) defined by MANO. The global rotations center ($\vec{t}_h$) depends on the estimated MANO shape parameters ($\vec{\beta}_h$). Using the estimated $\vec{\theta}_{hr}$ and $\vec{t}_h$, we transform $\vec{x}$ to the canonical hand pose (*i.e.,* the global rotation equals to zero):

$$\vec{x}_{hc} = \exp(\vec{\theta}_{hr})^{-1}(\vec{x} - \vec{t}_h) + \vec{t}_h, \tag{3}$$

where $\exp(\cdot)$ denotes the transformation from the axis-angle representation to the rotation matrix using the *Rodrigues formula*. Then, we concatenate $\vec{x}$ and $\vec{x}_{hc}$ and feed them to the hand SDF decoder and predict its signed distance to the hand:

$$\mathrm{SDF}_h(\vec{x}) = f_h(\vec{I}, [\vec{x},\ \vec{x}_{hc}]), \tag{4}$$

where f_h denotes the hand SDF decoder and $\vec{I}$ denotes image features extracted from the ResNet backbone. Benefiting from this formulation, the hand SDF encoder is aware of $\vec{x}$ in the canonical hand pose and can focus on learning the hand shape. Similarly, by estimating the object pose, we obtain the object translation $\vec{t}_o$ and transform $\vec{x}$ to the canonical object pose:

$$\vec{x}_{oc} = \vec{x} - \vec{t}_o. \tag{5}$$

Then, we concatenate $\vec{x}$ and $\vec{x}_{oc}$ and feed them to the object SDF decoder and predict its signed distance to the object:

$$\mathrm{SDF}_o(\vec{x}) = f_o(\vec{I}, [\vec{x},\ \vec{x}_{oc}]), \tag{6}$$

where f_o denotes the object SDF decoder. By feeding x_{oc} into f_o, the object SDF decoder can focus on learning the object shape in its canonical pose.

To train $\mathrm{SDF}_h(\vec{x})$ and $\mathrm{SDF}_o(\vec{x})$ we minimize L1 distance between predicted signed distances and corresponding ground-truth signed distances for sampled 3D points and training images. The resulting loss is the summation of L_{rec_h} and L_{rec_o}:

$$L_{rec} = \lambda_{rec_h} L_{rec_h} + \lambda_{rec_o} L_{rec_o}, \tag{7}$$

where L_{rec_h} and L_{rec_o} optimize $\mathrm{SDF}_h(\vec{x})$ and $\mathrm{SDF}_o(\vec{x})$, respectively. We set λ_{rec_h} and λ_{rec_o} to 5×10^{-1}. In summary, we train our model in an end-to-end fashion by minimizing the sum of losses introduced above:

$$L = L_{hand} + L_{obj} + L_{rec}. \tag{8}$$

Given the trained SDF networks, the hand and object surfaces are implicitly defined by the zero-level set of $\mathrm{SDF}_h(\vec{x})$ and $\mathrm{SDF}_o(\vec{x})$. We generate hand and object meshes using the Marching Cubes algorithm [35] at test time.

4 Experiments

In this section, we present a detailed evaluation of our proposed method. We introduce benchmarks in Section 4.1 and describe our evaluation metrics and implementation details in Sectoins 4.2-4.3. We then present hand-only ablations and hand-object experiments on the ObMan benchmark in Sections 4.5 and 4.4 respectively. Finally, we present experimental results for the DexYCB benchmark in Section 4.6. In the appendix, we illustrate our network architecture in Section A and provide more implementation details in Section B. We also show additional qualitative results in Section C.

4.1 Benchmarks

ObMan benchmark [21]. ObMan contains synthetic images and corresponding 3D meshes for a wide range of hand-object interactions with varying hand poses and objects. For training, we follow [26, 45] and discard meshes that contain too many double sided triangles, obtaining 87,190 samples. For each sample, we normalize the hand mesh and the object mesh so that they fit inside a unit cube and sample 40,000 points. At test time, we report results on 6285 samples following [21, 26].
DexYCB benchmark [6]. With 582K grasping frames for 20 YCB objects, DexYCB is currently the largest real benchmark for hand-object reconstruction. Following [33], we only consider right-hand samples and use the official "S0" split. We filter out the frames for which the minimum distance between the hand mesh and the object mesh is larger than 5 mm. We also normalize the hand mesh and the object mesh to a unit cube and sample 40,000 points to generate SDF training samples for DexYCB. As a result, we obtain 148,415 training samples and 29,466 testing samples.

4.2 Evaluation metrics

The output of our model is structured, and a single metric does not fully capture performance. Therefore, we employ different metrics to evaluate our method. Please see Section B in the appendix for more details.
Hand shape error ($\mathbf{H_{se}}$). We follow [26, 45] and evaluate the chamfer distance between reconstructed and ground-truth hand meshes to reflect hand reconstruction accuracy. Since the scale of the hand and the translation are ambiguous in monocular images, we optimize the scale and translation to align the reconstructed mesh with the ground-truth and sample 30,000 points from both meshes to calculate the chamfer distance. $\mathrm{H_{se}}$ (cm^2) is the median chamfer distance over the entire test set.
Hand validity error ($\mathbf{H_{ve}}$). Following [19, 76] we perform *Procrustes analysis* by optimizing the scale, translation and global rotation with regard to the ground-truth. We report $\mathrm{H_{ve}}$ (cm^2), the chamfer distance after alignment.

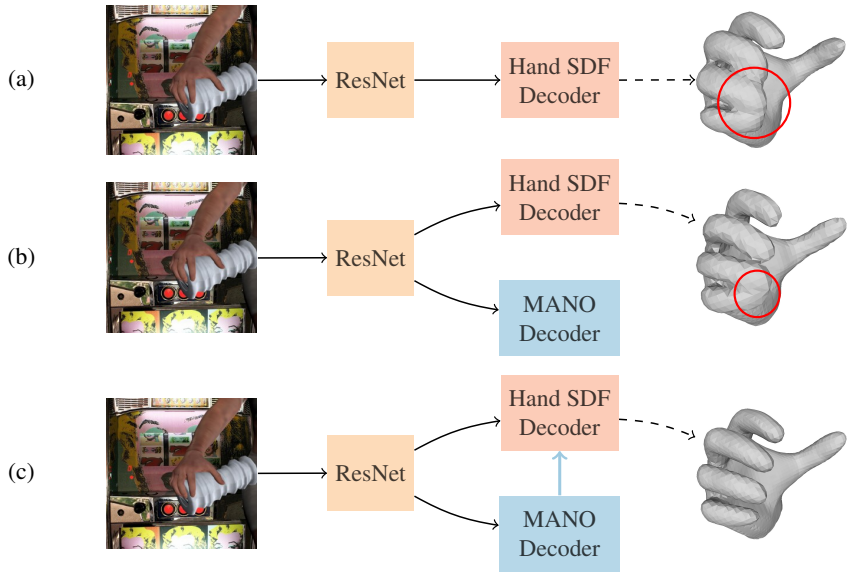


Fig. 3: Three baseline models for hand-only ablation experiments. Dashed arrows denote the Marching Cubes algorithm [35] used at test time.

Object shape error ($\mathbf{O_{se}}$). We reuse the optimized hand scale and translation from the computation of $\mathrm{H_{se}}$ to transform the reconstructed object mesh, following [26]. We follow the same process described for $\mathrm{H_{se}}$ to compute $\mathrm{O_{se}}$ (cm^2).
Hand joint error ($\mathbf{H_{je}}$). To measure the hand pose accuracy, we compute the mean join error (cm) relative to the hand wrist joint over 21 joints following [75].
Object translation error ($\mathbf{O_{te}}$). As we mention in Section 3.2, we predict the position of the object centroid relative to the hand wrist. We compute the L2 distance (cm) between the estimated object centroid and its ground-truth to report $\mathrm{O_{te}}$.
Contact ratio ($\mathbf{C_r}$). Following [26], we report the ratio of samples for which the interpenetration depth between the hand and the object is larger than zero.
Penetration depth. ($\mathbf{P_d}$). We compute the maximum of the distances (cm) from the hand mesh vertices to the object's surface similarly to [21, 26].
Intersection volume ($\mathbf{I_v}$). Following [21], we voxelize the hand and the object using a voxel size of 0.5 cm and compute their intersection volume (cm^3).

4.3 Implementation details

We use ResNet-18 [22] as a backbone to extract features from input images of size 256×256. To construct volumetric heatmaps, we employ three deconvolution layers to consecutively upsample feature maps from 8×8 to 64×64 and set the resolution of volumetric heatmaps to $64 \times 64 \times 64$. Please see the network architecture of the SDF decoder in Section A in the appendix. To train hand and object SDF decoders, we randomly sample 1,000 3D points (500 positive points outside the shape and 500

Table 1: Hand-only ablation experiments using 87K ObMan training samples.

Models	$H_{se}\downarrow$	$H_{ve}\downarrow$	$H_{je}\downarrow$
(a)	0.128	0.113	-
(b)	0.126	0.112	**1.18**
(c)	**0.124**	**0.109**	1.20
(c*)	0.101	0.087	-

Table 2: Hand-only ablation experiments using 30K ObMan training samples.

Models	$H_{se}\downarrow$	$H_{ve}\downarrow$	$H_{je}\downarrow$
(a)	0.183	0.160	-
(b)	0.176	0.156	**1.23**
(c)	**0.168**	**0.147**	1.27
(c*)	0.142	0.126	-

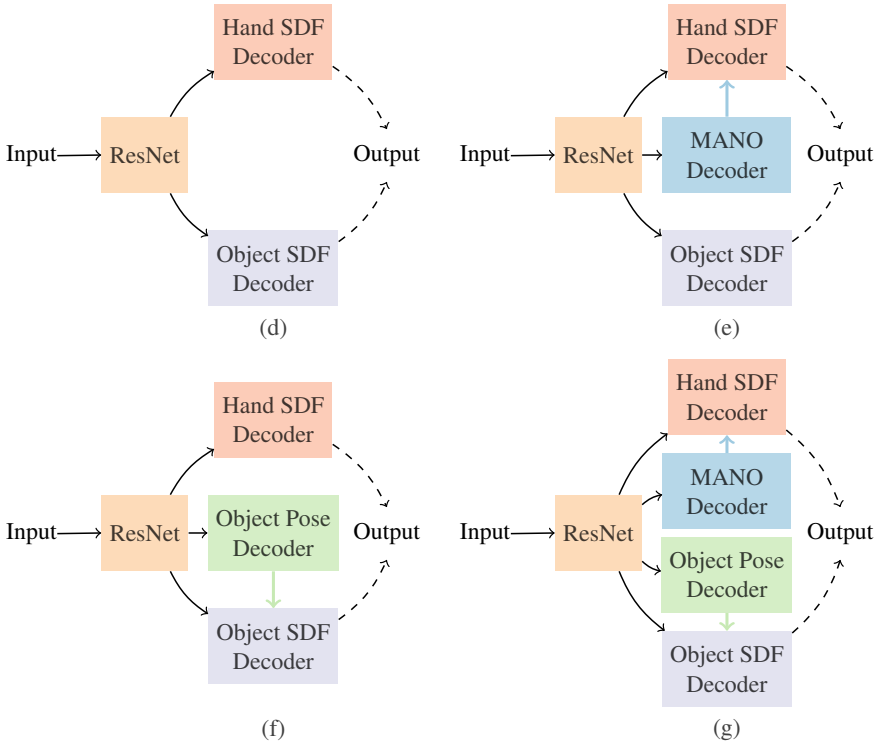


Fig. 4: Four models for hand-object ablation experiments. Dashed arrows denote the Marching Cubes algorithm [35] used at test time.

negative points inside the shape) for the hand and the object, respectively. We detail our data augmentation strategies used during training in Section B in the appendix. We train our model with the Adam optimizer [27] with a batch size of 256. We set the initial learning rate to 1×10^{-4} and decay it by half every 600 epoch on ObMan and every 300 epoch on DexYCB. The total number of training epochs is 1600 for ObMan and 800 for DexYCB, which takes about 90 hours on four NVIDIA 1080 Ti GPUs.

Table 3: Hand-object ablation experiments using 87K ObMan training data.

Models	$\mathrm{H_{se}} \downarrow$	$\mathrm{H_{ve}} \downarrow$	$\mathrm{O_{se}} \downarrow$	$\mathrm{H_{je}} \downarrow$	$\mathrm{O_{te}} \downarrow$	$\mathrm{C_r}$	$\mathrm{P_d}$	$\mathrm{I_v}$
(d)	0.140	0.124	4.09	-	-	90.3%	0.50	1.51
(e)	**0.131**	**0.114**	4.14	**1.12**	-	94.7%	0.58	2.00
(f)	0.148	0.130	**3.36**	-	**3.29**	92.5%	0.57	2.26
(g)	0.136	0.121	3.38	1.27	3.29	95.5%	0.66	2.81
(g^*)	0.111	0.093	2.11	-	-	94.5%	0.76	3.87

Table 4: Comparison with previous state-of-the-art methods on ObMan.

Methods	$\mathrm{H_{se}} \downarrow$	$\mathrm{H_{ve}} \downarrow$	$\mathrm{O_{se}} \downarrow$	$\mathrm{H_{je}} \downarrow$	$\mathrm{O_{te}} \downarrow$	$\mathrm{C_r}$	$\mathrm{P_d}$	$\mathrm{I_v}$
Hasson *et al.* [21]	0.415	0.383	3.60	**1.13**	-	94.8%	1.20	6.25
Karunratanakul *et al.* [26]-1De	0.261	0.246	6.80	-	-	5.63%	0.00	0.00
Karunratanakul *et al.* [26]-2De	0.237	-	5.70	-	-	69.6%	0.23	0.20
Ours (g)	**0.136**	**0.121**	**3.38**	1.27	**3.29**	95.5%	0.66	2.81

4.4 Hand-only experiments on ObMan

To validate the effectiveness of our method, we first conduct hand-only ablation experiments on ObMan. To this end, as shown in Figure 3, we first build three types of baseline models. The baseline model (a) directly employs the hand SDF decoder to learn $\mathrm{SDF}_h(\vec{x})$ from backbone features, which often results in a blurred reconstructed hand. The baseline model (b) trains the hand SDF decoder and the MANO network jointly and achieves better results. However, the reconstructed hand still suffers from ill-delimited outlines, which typically result in finger merging issues, illustrated in the second and third columns of Figure 5. Compared with the baseline model (b), the baseline model (c) further uses the estimated MANO parameters to transform sampled 3D points into the canonical hand pose, which helps disentangle the hand shape learning from the hand pose learning. As result, the hand SDF decoder can focus on learning the geometry of the hand and reconstruct a clear hand. The model (c^*) uses ground-truth hand poses, which is the upper-bound of our method. Tables 1 and 2 present quantitative results for these four models. In Table 1, we present our results using all ObMan training samples and observe that the baseline model (c) has the lowest $\mathrm{H_{se}}$ and $\mathrm{H_{ve}}$, which indicates that it achieves the best hand reconstruction quality. The baseline model (c) can also perform hand pose estimation well and reduce the joint error to 1.2 cm. It shows that the model (c) can transform $\vec{x}$ to the hand canonical pose well with reliable $\vec{\theta}_{hr}$ and $\vec{t}_h$ and benefit the learning of the hand SDFs. In Figure 5, we also visualize results obtained from different models and observe that our method can produce more precise hands even under occlusions. To check whether our method can still function well when the training data is limited, we randomly choose 30K samples to train these three models and summarize our results in Table 2. We observe that the advantage of the model (c) is more obvious using less training data. When compared with the model (a), our method can achieve more than 8% improvement in $\mathrm{H_{se}}$ and $\mathrm{H_{ve}}$, which further validates the effectiveness of our approach.

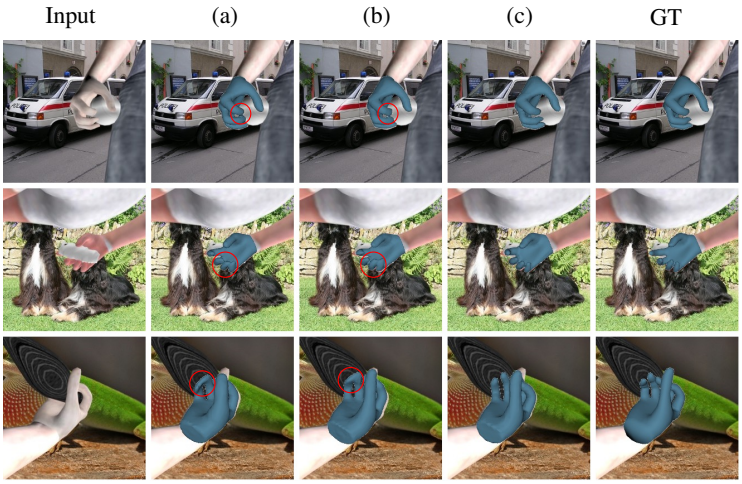


Fig. 5: Qualitative comparison of hand reconstructions between different hand-only baseline models on ObMan (87K training samples).

4.5 Hand-object experiments on ObMan

Given promising results for hand-only experiments, we next validate our approach for the task of hand-object reconstruction. As shown in Figure 4, we first build four baseline models. The baseline model (d) directly uses the hand and the object decoder to learn SDFs. Compared with the model (d), the model (e) estimates MANO parameters for the hand and uses it to improve the learning the hand SDF decoder. The model (f) estimates the object pose and uses the estimated pose to learn the object SDF decoder. The model (g) combines models (e) and (f) and uses estimated hand and object poses to improve the learning of the hand SDFs and the object SDFs, respectively. The model (g^*) is trained with ground-truth hand poses and object translations, which serves as the upper-bound for our method. We summarize our experimental results for these five models in Table 3. Compared with the baseline model (d), the model (e) achieves a 6.4% and 8.8% improvement in $\mathrm{H_{se}}$ and $\mathrm{H_{ve}}$, respectively. It shows that embedding hand prior knowledge and aligning hand poses to the canonical pose can improve learning the hand SDFs. By comparing the model (f) with the baseline model (d), we align object poses to their canonical poses using estimated object pose parameters and greatly reduce $\mathrm{O_{se}}$ from 4.09 cm^2 to 3.36 cm^2. Finally, our full model (g) combines the advantages of models (e) and (f) and can produce high-quality hand meshes and object meshes. In Table 4, we compare our method against previous state-of-the-art methods and show that our approach outperforms previous state-of-the-art methods [21, 26] by a significant margin. When we take a closer look at metrics ($\mathrm{C_r}$, $\mathrm{P_d}$, $\mathrm{I_v}$) that reflect hand-object interactions, we can observe that the reconstructed hand and the reconstructed object from our model are in contact with each other in more than 95.5% of test samples. Compared with the SDF method [26], our method encourages the contact between the hand mesh and the object mesh. Compared with the MANO-based method [21],

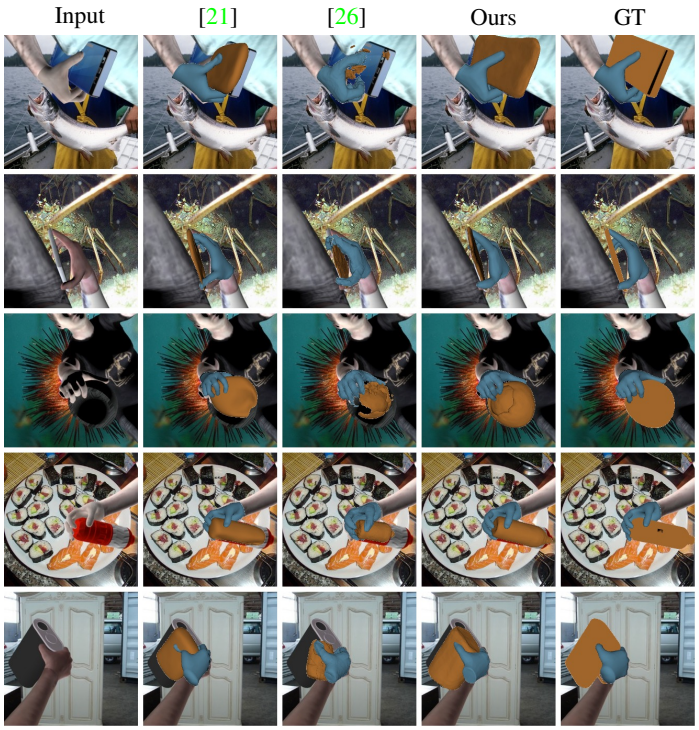


Fig. 6: Qualitative comparison between different types of methods in hand-object experiments on ObMan. Compared with recent methods [21, 26], our approach produces more precise reconstructions both for the hands and objects.

Table 5: Comparison with previous state-of-the-art methods on DexYCB.

Method	$H_{se}\downarrow$	$H_{ve}\downarrow$	$O_{se}\downarrow$	$H_{je}\downarrow$	$O_{te}\downarrow$	C_r	P_d	I_v
Hasson *et al.* [21]	0.785	0.594	4.4	2.0	-	95.8%	1.32	7.67
Karunratanakul *et al.* [26]	0.741	0.532	5.8	-	-	96.7%	0.83	1.34
Ours (g)	**0.523**	**0.375**	**3.5**	**1.9**	**2.7**	96.1%	0.71	3.45

the penetration depth (P_d) and intersection volume (I_v) of our model is much lower, which suggests that our method can produce more detailed hand-object interactions. In Figure 6, we also visualize reconstruction results from different methods. Compared to previous methods, our model can produce more realistic joint reconstruction results even for objects with thin structures. We include more qualitative analysis on ObMan in Section C in the appendix.

4.6 Hand-object experiments on DexYCB

To validate our method on real data, we next present experiments on the DexYCB benchmark and compare our results to the state of the art. We summarize our experimental results in Table 5. Compared with previous methods, we achieve a 29.4% im-

Fig. 7: Qualitative results of our model on the DexYCB benchmark. Our model produces convincing 3D hand-and-object reconstruction results in the real-world setting.

provement in H_{se} and a 20.5% improvement in O_{se}, which shows that our method improves both the hand and object reconstruction accuracy. The hand-object interaction metrics for DexYCB also indicate that our method works well for real images. Figure 7 illustrates qualitative results of our method on the DexYCB benchmark. We can observe that our method can accurately reconstruct hand shapes under different poses and a wide range of real-world objects. Please see more qualitative results on DexYCB in Section C in the appendix.

5 Conclusion

In this work, we combine advantages of parametric mesh models and SDFs for the task of a joint hand-object reconstruction. To embed prior knowledge into SDFs and to increase the learning efficiency, we propose to disentangle the shape learning and pose learning for both the hand and the object. Then, we align SDF representations with respect to estimated poses and enable learning of more accurate shape estimation. Our model outperforms previous state-of-the-art methods by a significant margin on main benchmarks. Our results also demonstrate significant improvements in visual quality.

Acknowledgements. This work was granted access to the HPC resources of IDRIS under the allocation AD011013147 made by GENCI. This work was funded in part by the French government under management of Agence Nationale de la Recherche as part of the "Investissements d'avenir" program, reference ANR19-P3IA-0001 (PRAIRIE 3IA Institute) and by Louis Vuitton ENS Chair on Artificial Intelligence.

References

[1] Baek, S., Kim, K.I., Kim, T.K.: Augmented skeleton space transfer for depth-based hand pose estimation. In: CVPR (2018)

[2] Baek, S., Kim, K.I., Kim, T.K.: Pushing the envelope for RGB-based dense 3D hand pose estimation via neural rendering. In: CVPR (2019)

[3] Ballan, L., Taneja, A., Gall, J., Gool, L.V., Pollefeys, M.: Motion capture of hands in action using discriminative salient points. In: ECCV (2012)

[4] Boukhayma, A., Bem, R.d., Torr, P.H.: 3D hand shape and pose from images in the wild. In: CVPR (2019)

[5] Cai, Y., Ge, L., Cai, J., Thalmann, N.M., Yuan, J.: 3D hand pose estimation using synthetic data and weakly labeled RGB images. TPAMI (2020)

[6] Chao, Y.W., Yang, W., Xiang, Y., Molchanov, P., Handa, A., Tremblay, J., Narang, Y.S., Van Wyk, K., Iqbal, U., Birchfield, S., et al.: DexYCB: A benchmark for capturing hand grasping of objects. In: CVPR (2021)

[7] Chen, L., Lin, S.Y., Xie, Y., Lin, Y.Y., Xie, X.: MVHM: A large-scale multi-view hand mesh benchmark for accurate 3D hand pose estimation. In: WACV (2021)

[8] Chen, X., Liu, Y., Ma, C., Chang, J., Wang, H., Chen, T., Guo, X., Wan, P., Zheng, W.: Camera-space hand mesh recovery via semantic aggregation and adaptive 2D-1D registration. In: CVPR (2021)

[9] Chen, X., Wang, G., Zhang, C., Kim, T.K., Ji, X.: SHPR-Net: Deep semantic hand pose regression from point clouds. IEEE Access (2018)

[10] Chen, X., Zheng, Y., Black, M.J., Hilliges, O., Geiger, A.: SNARF: Differentiable forward skinning for animating non-rigid neural implicit shapes. In: ICCV (2021)

[11] Chen, Z., Zhang, H.: Learning implicit fields for generative shape modeling. In: CVPR (2019)

[12] Choy, C.B., Xu, D., Gwak, J., Chen, K., Savarese, S.: 3D-R2N2: A unified approach for single and multi-view 3D object reconstruction. In: ECCV (2016)

[13] Cootes, T.F., Taylor, C.J., Cooper, D.H., Graham, J.: Active shape models-their training and application. CVIU (1995)

[14] Deng, B., Lewis, J.P., Jeruzalski, T., Pons-Moll, G., Hinton, G., Norouzi, M., Tagliasacchi, A.: NASA: Neural articulated shape approximation. In: ECCV (2020)

[15] Doosti, B., Naha, S., Mirbagheri, M., Crandall, D.: HOPE-Net: A graph-based model for hand-object pose estimation. In: CVPR (2020)

[16] Groueix, T., Fisher, M., Kim, V.G., Russell, B.C., Aubry, M.: A papier-mâché approach to learning 3D surface generation. In: CVPR (2018)

[17] Hamer, H., Gall, J., Weise, T., Van Gool, L.: An object-dependent hand pose prior from sparse training data. In: CVPR (2010)

[18] Hamer, H., Schindler, K., Koller-Meier, E., Van Gool, L.: Tracking a hand manipulating an object. In: ICCV (2009)

[19] Hampali, S., Rad, M., Oberweger, M., Lepetit, V.: HOnnotate: A method for 3D annotation of hand and object poses. In: CVPR (2020)

[20] Hasson, Y., Tekin, B., Bogo, F., Laptev, I., Pollefeys, M., Schmid, C.: Leveraging photometric consistency over time for sparsely supervised hand-object reconstruction. In: CVPR (2020)

[21] Hasson, Y., Varol, G., Tzionas, D., Kalevatykh, I., Black, M.J., Laptev, I., Schmid, C.: Learning joint reconstruction of hands and manipulated objects. In: CVPR (2019)
[22] He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: CVPR (2016)
[23] Heap, T., Hogg, D.: Towards 3D hand tracking using a deformable model. In: FG (1996)
[24] Iqbal, U., Molchanov, P., Gall, T.B.J., Kautz, J.: Hand pose estimation via latent 2.5D heatmap regression. In: ECCV (2018)
[25] Karunratanakul, K., Spurr, A., Fan, Z., Hilliges, O., Tang, S.: A skeleton-driven neural occupancy representation for articulated hands. In: 3DV (2021)
[26] Karunratanakul, K., Yang, J., Zhang, Y., Black, M.J., Muandet, K., Tang, S.: Grasping Field: Learning implicit representations for human grasps. In: 3DV (2020)
[27] Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014)
[28] Kokic, M., Kragic, D., Bohg, J.: Learning to estimate pose and shape of hand-held objects from RGB images. In: IROS (2019)
[29] Kulon, D., Güler, R.A., Kokkinos, I., Bronstein, M., Zafeiriou, S.: Weakly-supervised mesh-convolutional hand reconstruction in the wild. In: CVPR (2020)
[30] Kulon, D., Wang, H., Güler, R.A., Bronstein, M.M., Zafeiriou, S.: Single image 3D hand reconstruction with mesh convolutions. In: BMVC (2019)
[31] Labbé, Y., Carpentier, J., Aubry, M., Sivic, J.: CosyPose: Consistent multi-view multi-object 6D pose estimation. In: ECCV (2020)
[32] Labbé, Y., Carpentier, J., Aubry, M., Sivic, J.: Single-view robot pose and joint angle estimation via render & compare. In: CVPR (2021)
[33] Li, K., Yang, L., Zhan, X., Lv, J., Xu, W., Li, J., Lu, C.: ArtiBoost: Boosting articulated 3D hand-object pose estimation via online exploration and synthesis. In: CVPR (2022)
[34] Li, Y., Wang, G., Ji, X., Xiang, Y., Fox, D.: DeepIM: Deep iterative matching for 6D pose estimation. In: ECCV (2018)
[35] Lorensen, W.E., Cline, H.E.: Marching Cubes: A high resolution 3D surface construction algorithm. TOG (1987)
[36] Mescheder, L., Oechsle, M., Niemeyer, M., Nowozin, S., Geiger, A.: Occupancy Networks: Learning 3D reconstruction in function space. In: CVPR (2019)
[37] Mildenhall, B., Srinivasan, P.P., Tancik, M., Barron, J.T., Ramamoorthi, R., Ng, R.: NeRF: Representing scenes as neural radiance fields for view synthesis. In: ECCV (2020)
[38] Moon, G., Chang, J.Y., Lee, K.M.: V2V-PoseNet: Voxel-to-voxel prediction network for accurate 3D hand and human pose estimation from a single depth map. In: CVPR (2018)
[39] Moon, G., Shiratori, T., Lee, K.M.: DeepHandMesh: A weakly-supervised deep encoder-decoder framework for high-fidelity hand mesh modeling. In: ECCV (2020)
[40] Mueller, F., Bernard, F., Sotnychenko, O., Mehta, D., Sridhar, S., Casas, D., Theobalt, C.: GANerated hands for real-time 3D hand tracking from monocular RGB. In: CVPR (2018)

[41] Mueller, F., Davis, M., Bernard, F., Sotnychenko, O., Verschoor, M., Otaduy, M.A., Casas, D., Theobalt, C.: Real-time pose and shape reconstruction of two interacting hands with a single depth camera. TOG (2019)
[42] Mundy, J.L.: Object recognition in the geometric era: A retrospective. In: Toward Category-Level Object Recognition, Lecture Notes in Computer Science (2006)
[43] Oikonomidis, I., Kyriazis, N., Argyros, A.A.: Full DOF tracking of a hand interacting with an object by modeling occlusions and physical constraints. In: ICCV (2011)
[44] Panteleris, P., Oikonomidis, I., Argyros, A.: Using a single RGB frame for real time 3D hand pose estimation in the wild. In: WACV (2018)
[45] Park, J.J., Florence, P., Straub, J., Newcombe, R., Lovegrove, S.: DeepSDF: Learning continuous signed distance functions for shape representation. In: CVPR (2019)
[46] Pavlakos, G., Zhou, X., Derpanis, K.G., Daniilidis, K.: Coarse-to-fine volumetric prediction for single-image 3D human pose. In: CVPR (2017)
[47] Peng, S., Jiang, C., Liao, Y., Niemeyer, M., Pollefeys, M., Geiger, A.: Shape As Points: A differentiable poisson solver. In: NeurIPS (2021)
[48] Qi, C.R., Su, H., Mo, K., Guibas, L.J.: PointNet: Deep learning on point sets for 3D classification and segmentation. In: CVPR (2017)
[49] Qian, N., Wang, J., Mueller, F., Bernard, F., Golyanik, V., Theobalt, C.: HTML: A parametric hand texture model for 3D hand reconstruction and personalization. In: ECCV (2020)
[50] Rehg, J.M., Kanade, T.: Visual tracking of high DOF articulated structures: an application to human hand tracking. In: ECCV (1994)
[51] Riegler, G., Ulusoy, A.O., Geiger, A.: OctNet: Learning deep 3D representations at high resolutions. In: CVPR (2017)
[52] Roberts, L.G.: Machine perception of three-dimensional solids. Ph.D. thesis, Massachusetts Institute of Technology (1963)
[53] Romero, J., Kjellström, H., Kragic, D.: Hands in Action: Real-time 3D reconstruction of hands in interaction with objects. In: ICRA (2010)
[54] Romero, J., Tzionas, D., Black, M.J.: Embodied Hands: Modeling and capturing hands and bodies together. TOG (2017)
[55] Saito, S., Huang, Z., Natsume, R., Morishima, S., Kanazawa, A., Li, H.: PIFu: Pixel-aligned implicit function for high-resolution clothed human digitization. In: ICCV (2019)
[56] Saito, S., Yang, J., Ma, Q., Black, M.J.: SCANimate: Weakly supervised learning of skinned clothed avatar networks. In: CVPR (2021)
[57] Spurr, A., Dahiya, A., Wang, X., Zhang, X., Hilliges, O.: Self-supervised 3D hand pose estimation from monocular RGB via contrastive learning. In: CVPR (2021)
[58] Sridhar, S., Mueller, F., Zollhöfer, M., Casas, D., Oulasvirta, A., Theobalt, C.: Real-time joint tracking of a hand manipulating an object from RGB-D input. In: ECCV (2016)
[59] Su, S.Y., Yu, F., Zollhöfer, M., Rhodin, H.: A-NeRF: Articulated neural radiance fields for learning human shape, appearance, and pose. In: NeurIPS (2021)
[60] Sun, X., Xiao, B., Wei, F., Liang, S., Wei, Y.: Integral human pose regression. In: ECCV (2018)

[61] Supančič, J.S., Rogez, G., Yang, Y., Shotton, J., Ramanan, D.: Depth-based hand pose estimation: methods, data, and challenges. IJCV (2018)
[62] Tekin, B., Bogo, F., Pollefeys, M.: H+O: Unified egocentric recognition of 3D hand-object poses and interactions. In: CVPR (2019)
[63] Tsoli, A., Argyros, A.A.: Joint 3D tracking of a deformable object in interaction with a hand. In: ECCV (2018)
[64] Tzionas, D., Gall, J.: 3D object reconstruction from hand-object interactions. In: ICCV (2015)
[65] Wang, J., Mueller, F., Bernard, F., Sorli, S., Sotnychenko, O., Qian, N., Otaduy, M.A., Casas, D., Theobalt, C.: RGB2Hands: Real-time tracking of 3D hand interactions from monocular RGB video. TOG (2020)
[66] Wang, N., Zhang, Y., Li, Z., Fu, Y., Liu, W., Jiang, Y.G.: Pixel2Mesh: Generating 3D mesh models from single RGB images. In: ECCV (2018)
[67] Wang, Y., Min, J., Zhang, J., Liu, Y., Xu, F., Dai, Q., Chai, J.: Video-based hand manipulation capture through composite motion control. TOG (2013)
[68] Xiang, Y., Schmidt, T., Narayanan, V., Fox, D.: PoseCNN: A convolutional neural network for 6D object pose estimation in cluttered scenes. In: RSS (2018)
[69] Xiong, F., Zhang, B., Xiao, Y., Cao, Z., Yu, T., Zhou, J.T., Yuan, J.: A2J: Anchor-to-joint regression network for 3D articulated pose estimation from a single depth image. In: ICCV (2019)
[70] Xu, Q., Wang, W., Ceylan, D., Mech, R., Neumann, U.: DISN: Deep implicit surface network for high-quality single-view 3D reconstruction. In: NeurIPS (2019)
[71] Yang, L., Zhan, X., Li, K., Xu, W., Li, J., Lu, C.: CPF: Learning a contact potential field to model the hand-object interaction. In: ICCV (2021)
[72] Ye, Y., Gupta, A., Tulsiani, S.: What's in your hands? 3D reconstruction of generic objects in hands. In: CVPR (2022)
[73] Zhang, H., Zhou, Y., Tian, Y., Yong, J.H., Xu, F.: Single depth view based real-time reconstruction of hand-object interactions. TOG (2021)
[74] Zhou, Y., Habermann, M., Xu, W., Habibie, I., Theobalt, C., Xu, F.: Monocular real-time hand shape and motion capture using multi-modal data. In: CVPR (2020)
[75] Zimmermann, C., Brox, T.: Learning to estimate 3D hand pose from single RGB images. In: ICCV (2017)
[76] Zimmermann, C., Ceylan, D., Yang, J., Russell, B., Argus, M., Brox, T.: FreiHAND: A dataset for markerless capture of hand pose and shape from single RGB images. In: ICCV (2019)

AlignSDF: Pose-Aligned Signed Distance Fields for Hand-Object Reconstruction

Appendix

In this appendix, we provide additional details for our experimental settings as well as qualitative results of our method. We first present details for our network architecture in Section A. Section B then provides additional implementation details for our training and evaluation procedures. Finally, we present and discuss additional qualitative results in Section C.

A Network Architecture

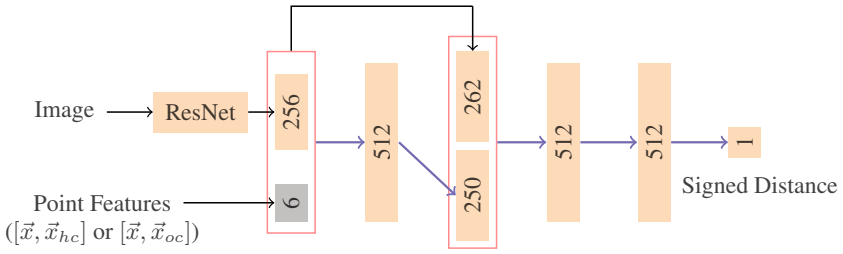


Fig. A.1: Network architecture used for our hand and object SDF decoders. Following [26], we also use five fully connected layers (marked in purple) for the SDF decoder. The number in the box denotes the dimension of features. $\vec{x}$ denotes the original 3D coordinate. $\vec{x}_{hc}$ and $\vec{x}_{oc}$ denote the transformed 3D coordinate in the hand and object canonical coordinate system, respectively.

Following previous works [21, 26], we use ResNet-18 [22] as our backbone network. To achieve a fair comparison with the previous method [26], as shown in Figure A.1, we also use five fully connected layers to estimate the signed distance from the query point to the hand surface or the object surface. The SDF decoder takes the 256-dimensional image features and 6-dimensional point features as inputs. The image features are extracted from the ResNet-18 backbone. Following Equation 3 and Equation 5 in our paper, we transform the original 3D point $\vec{x}$ into its counterpart $\vec{x}_{hc}$ in the hand canonical coordinate system or its counterpart $\vec{x}_{oc}$ in the object canonical coordinate system. Then, we construct point features by concatenating $\vec{x}$ and $\vec{x}_{hc}$ for the hand SDF decoder or by concatenating $\vec{x}$ and $\vec{x}_{oc}$ for the object SDF decoder.

B Training and Evaluation

We train all of our models with the following data augmentation. We randomly rotate the input image and 3D points in the camera coordinate system. We empirically find that

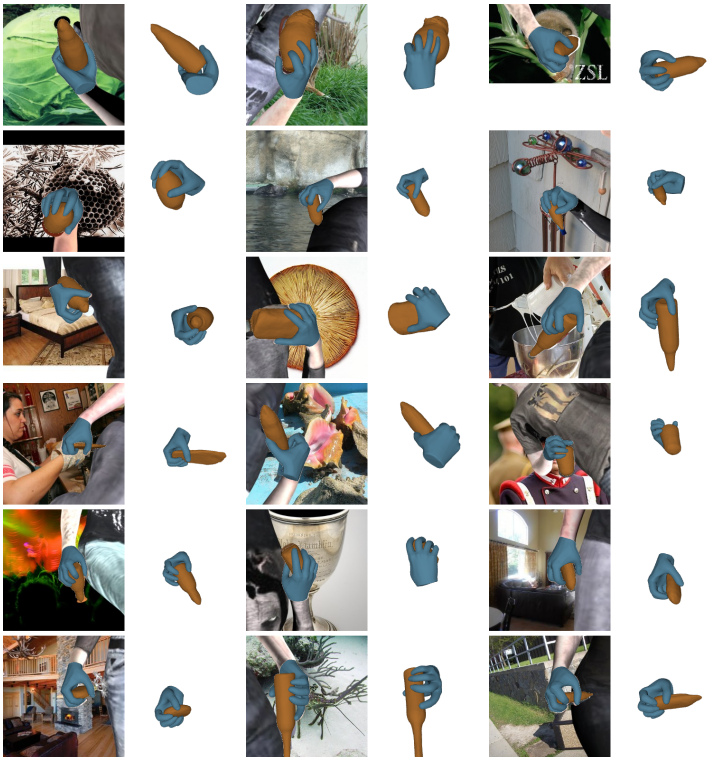


Fig. C.1: Qualitative results of our method on the ObMan [21] benchmark. Our method can produce convincing 3D reconstruction results even in cluttered scenes.

this data augmentation can boost the performance for 3D reconstruction. We randomly augment training samples via $[-45°, 45°]$ rotation for our experiments on ObMan [21] or $[-15°, 15°]$ rotation for our experiments on DexYCB [6].

We set the hand wrist joint defined by MANO [54] as the origin of our coordinate system. In training, we use a fixed scaling factor to scale all negative points (*i.e.,* points that lie in the hand or object mesh) across the dataset within a unit cube. This results in a scaling factor of 7.02 and 6.21 on ObMan and DexYCB, respectively.

To measure the physical quality of our joint reconstruction, we report Contact Ratio (C_r), Penetration Depth (P_d) and Intersection Volume (I_v). We use the trimesh library to detect whether there exists a collision between the hand mesh and the object mesh and compute the max penetration depth between two meshes. We follow the same process as [25, 26] to compute I_v.

C Qualitative results

We present additional qualitative results on ObMan [21] in Figure C.1 and DexYCB [6] in Figure C.2. We also study failure cases on DexYCB in Figure C.3. From Figure C.1,

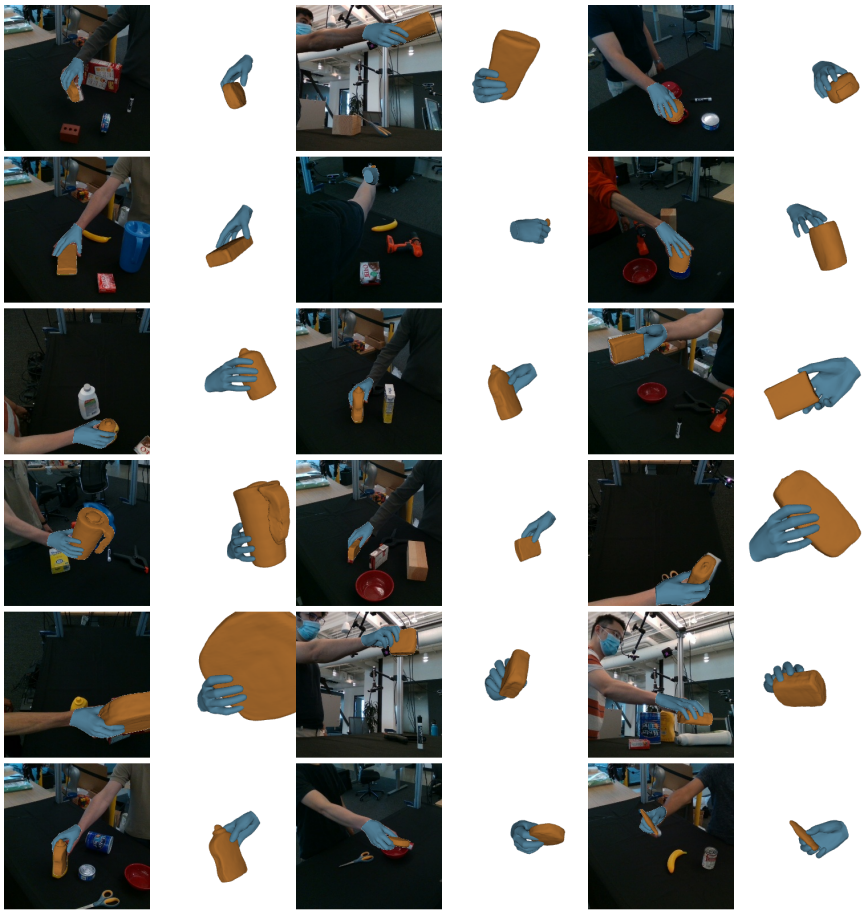
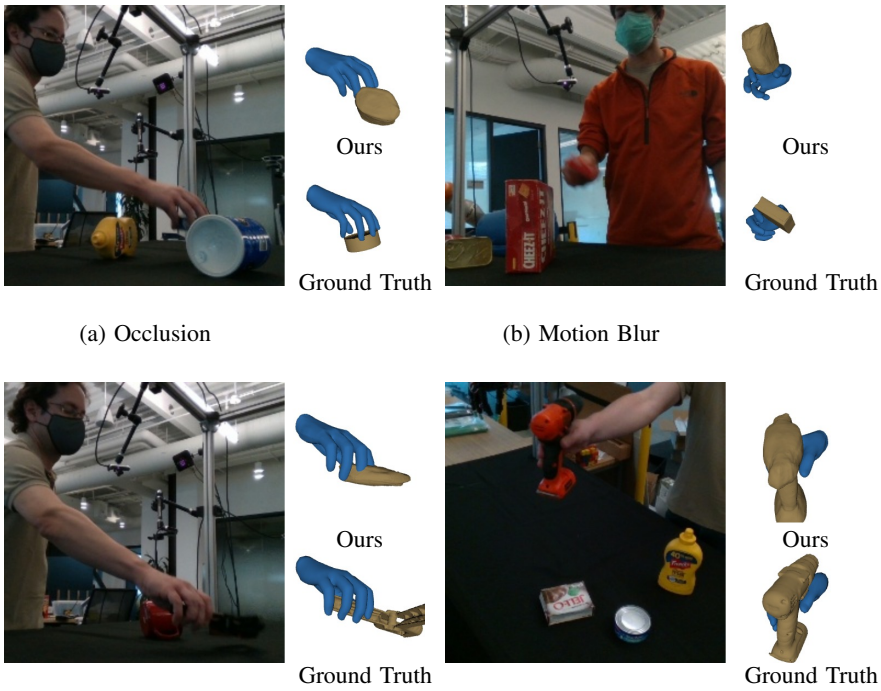


Fig. C.2: Qualitative results of our method on the DexYCB [6] benchmark. Our method can also produce realistic 3D reconstruction results for real scenes.

we observe that our method can deal with a wide range of objects and recovers detailed interactions between the hand and the object. In Figure C.2 we show qualitative results of our method for real images from the DexYCB benchmark. We can see that our method can reconstruct objects of different sizes and often achieve the excellent reconstruction of hands and objects.

While our method advances the state of the art accuracy by a significant margin, it still does not achieve satisfactory performance in some cases. In Figure C.3 we show four typical failure cases on DexYCB. As shown in Figure C.3(a), when the hand or the object is heavily occluded, our method sometimes cannot make robust predictions. In Figure C.3(b), we show that motion blur in input images might also disturb 3D reconstruction results. As shown in Figure C.3(c, d), the recovery of thin structures and objects with complex shapes remains challenging. To deal with these issues, future



(a) Occlusion (b) Motion Blur

(c) Thin Structures (d) Complex Details

Fig. C.3: Failure cases of our method on the DexYCB [6] benchmark.

works could leverage the temporary information from videos to filter input noise and gather more details about 3D scenes.