Abstract

Signed distance functions (SDFs) is an attractive framework that has recently shown promising results for 3D shape reconstruction from images. SDFs seamlessly generalize to different shape resolutions and topologies but lack explicit modelling of the underlying 3D geometry. In this work, we exploit the hand structure and use it as guidance for SDF-based shape reconstruction. In particular, we address reconstruction of hands and manipulated objects from monocular RGB images. To this end, we estimate poses of hands and objects and use them to guide 3D reconstruction. More specifically, we predict kinematic chains of pose transformations and align SDFs with highly-articulated hand poses. We improve the visual features of 3D points with geometry alignment and further leverage temporal information to enhance the robustness to occlusion and motion blurs. We conduct extensive experiments on the challenging ObMan and DexYCB benchmarks and demonstrate significant improvements of the proposed method over the state of the art.

1. Introduction

Understanding how hands interact with objects is becoming increasingly important for widespread applications, including virtual reality, robotic manipulation and human-computer interaction. Compared to 3D estimation of sparse hand joints [24, 38, 51, 53, 67], joint reconstruction of hands and object meshes [11, 18, 21, 26, 62] provides rich information about hand-object interactions and has received increased attention in recent years.

To reconstruct high-quality meshes, some recent works [9, 17, 61] explore multi-view image inputs. Multi-view images, however, are less common both for training and testing scenarios. In this work, we focus on a more practical and user-friendly setting where we aim to reconstruct hand and object meshes from monocular RGB images. Given the ill-posed nature of the task, many existing methods [7, 19, 21, 54, 62] employ parametric mesh models (e.g., MANO [46]) to impose prior knowledge and reduce ambiguities in 3D hand reconstruction. MANO hand meshes, however, have relatively limited resolution and can be suboptimal for the precise capture of hand-object interactions.

To reconstruct detailed hand and object meshes, another line of efforts [11, 26] employ signed distance functions (SDFs). Grasping Field [26] makes the first attempt to model hand and object surfaces using SDFs. However, it does not explicitly associate 3D geometry with image cues and has no prior knowledge incorporated in SDFs, leading to unrealistic meshes. AlignSDF [11] proposes to align SDFs with respect to global poses (i.e., the hand wrist transformation and the
object translation) and produces improved results. However, it is still challenging to capture geometric details for more complex hand motions and manipulations of diverse objects, which involve the articulation of multiple fingers.

To address limitations of prior works, we propose a geometry-driven SDF (gSDF) method that encodes strong pose priors and improves reconstruction by disentangling pose and shape estimation (see Figure 1). To this end, we first predict sparse 3D hand joints from images and derive full kinematic chains of local pose transformations from joint locations using inverse kinematics. Instead of only using the global pose as in [11], we optimize SDFs with respect to poses of all the hand joints, which leads to a more fine-grained alignment between the 3D shape and articulated hand poses. In addition, we project 3D points onto the image plane to extract geometry-aligned visual features for signed distance prediction. The visual features are further refined with spatio-temporal contexts using a transformer model to enhance the robustness to occlusions and motion blurs.

We conduct extensive ablation experiments to show the effectiveness of different components in our approach. The proposed gSDF model greatly advances state-of-the-art accuracy on the challenging ObMan and DexYCB benchmarks. Our contributions can be summarized in three-fold: (i) To embed strong pose priors into SDFs, we propose to align the SDF shape with its underlying kinematic chains of pose transformations, which reduces ambiguities in 3D reconstruction. (ii) To further reduce the misalignment induced by inaccurate pose estimations, we propose to extract geometry-aligned local visual features and enhance the robustness with spatio-temporal contexts. (iii) We conduct comprehensive experiments to show that our approach outperforms state-of-the-art results by a significant margin.

2. Related Work

This paper focuses on jointly reconstructing hands and hand-held objects from RGB images. In this section, we first review previous works on the 3D hand pose and shape estimation. We then discuss relevant works on the joint reconstruction of hands and objects.

3D hand pose and shape estimation. The topic of 3D hand pose estimation has received widespread attention since the 90s [23, 45] and has seen significant progress in recent years [31, 65]. Methods which take RGB images as input [24, 36, 38, 39, 48, 50, 51, 53, 59, 67] often estimate sparse 3D hand joint locations from visual data using well-designed deep neural networks. Though these methods can achieve high estimation accuracy, their outputs of 3D sparse joints provide limited information about the 3D hand surface, which is critical in AR/VR applications. Following the introduction of the anthropomorphic parametric hand mesh model MANO [46], several works [2, 5, 10, 18, 29, 30, 32, 34, 40, 57] estimate the MANO hand shape and pose parameters to recover the full hand surface. However, MANO has a limited mesh resolution and cannot produce fine surface details. Neural implicit functions [13, 25] have the potential to reconstruct more realistic high-resolution hand surfaces [12, 37, 42]. In this work, we combine the advantages of sparse, parametric and implicit modeling. We predict sparse 3D joints accurately from images and estimate the MANO parameters using inverse kinematics. We then optimize neural implicit functions with respect to underlying kinematic structures and reconstruct realistic meshes.

3D hand and object reconstruction. Joint reconstruction of hand and object meshes provides a more comprehensive view about how hands interact with manipulated objects in the 3D space and has received more attention in the past few years. Previous works often rely on multi-view correspondence [3, 9, 17, 41, 58, 61] or additional depth information [15, 16, 49, 55, 56] to approach this task. In this work, we focus on a more challenging setting and perform a joint reconstruction from a monocular RGB image. Given the ill-posed nature of this problem, many works [7, 18–21, 54, 60, 62] deploy MANO, which encodes hand prior knowledge learned from hand scans, to reconstruct hand meshes. To further simplify the object reconstruction task, several works [18, 60, 62] make a strong assumption that the ground-truth object model is available at test time. Our work and some previous efforts [11, 21, 26] relax this assumption and assume unknown object models. Hasson et al. [21] employ a differentiable MANO layer to estimate the hand shape and AtlasNet [14] to reconstruct the manipulated object. However, both MANO and AtlasNet can only produce meshes of limited resolution, which prevents the modeling of detailed contacts between hands and objects. To generate more detailed surfaces, Karunratanakul et al. [26] introduce grasping fields and propose to use SDFs to reconstruct both hand and object meshes. However, such a model-free approach does not capture any prior knowledge about hands or objects, which can lead to predicting unrealistic 3D geometry. To mitigate this, Ye et al. [63] propose to use hand poses estimated from an off-the-shelf model to help reconstruct the hand-held object mesh. The main difference with our work is that we jointly reconstruct hand meshes and object meshes using our proposed model, which is more challenging. Also, in addition to using hand poses to help capture the object shapes, we predict object poses and show their benefits for SDF-based object reconstruction. Another work AlignSDF [11] optimizes SDFs with respect to estimated hand-object global poses and encodes pose priors into SDFs. In addition to using global poses as a guide for SDFs, we propose to learn SDFs from the full kinematic chains of local pose transformations, and achieve a more precise alignment between the 3D shape and the underlying poses. To further handle hard cases induced by occlusion or motion blur where pose estimations are inaccurate, we leverage
a transformer to accumulate corresponding image features from multiple frames and benefit the geometry recovery.

3. Method

This section presents our geometry-driven SDF (gSDF) method for 3D hand and object reconstruction from monocular RGB images. We aim to learn two signed distance functions $SDF_{hand}$ and $SDF_{obj}$ to implicitly represent 3D shapes for the hand and the object. The $SDF_{hand}$ and $SDF_{obj}$ map a query 3D point $x \in \mathbb{R}^3$ to a signed distance from the hand surface and object surface, respectively. The Marching Cubes algorithm [33] can thus be employed to reconstruct the hand and the object from $SDF_{hand}$ and $SDF_{obj}$.

3.1. Overview of gSDF

Figure 2 illustrates the overview of our gSDF reconstruction approach. Given an image $I_t$, we extract two types of features to predict the signed distance for each query point $x$, namely kinematic features and visual features.

The kinematic feature encodes the position of $x$ under the coordinate system of the hand or the object, which can provide strong pose priors to assist SDF learning. Since the feature is based on canonical hand and object poses, it helps to disentangle shape learning from pose learning.

The existing work [63] proposes to use hand poses for reconstructing object meshes but does not consider using pose priors to reconstruct hand meshes. Another work [11] only deploys coarse geometry in terms of the hand wrist object locations, which fails to capture fine-grained details. In this work, we aim to strengthen the kinematic feature with geometry transformation of $x$ to poses of all the hand joints (see Figure 3) for both the hand and the object reconstruction. However, it is challenging to directly predict hand pose parameters [6, 28, 66]. To improve the hand pose estimation, we propose to first predict sparse 3D joint locations $j_h$ from the image and then use inverse kinematics to derive pose transformations $\theta_h$ from the predicted joints. In this way, we are able to obtain kinematic features $e_h$ and $e_o$ for the hand and the object respectively.

The visual feature encodes the visual appearance for the point $x$ to provide more shape details. Prior works [11, 26] use the same global visual feature for all the points, e.g., averaging the feature map of a SDF feature encoder on the spatial dimension. Such global visual features suffers from imprecise geometry alignment between a point and its visual appearance. To alleviate the limitation, inspired by [47], we apply the geometry transformation to extract aligned local visual features. Moreover, to address hard cases with occlusions and motion blur in a single image $I_t$, we propose to enhance the local visual feature with its temporal contexts from videos using a spatio-temporal transformer. We denote the local visual feature of a point as $e_v$. Finally, we concatenate the kinematic feature and local visual feature to predict the signed distance for $x$:

$$
SDF_{hand}(x) = f_h([e_v; e_h]),
SDF_{object}(x) = f_o([e_v; e_o]),
$$

(1)

where $f_h$ and $f_o$ are the hand SDF decoder and the object SDF decoder respectively.

In the following, we first present the proposed geometry-driven kinematic feature and visual feature encodings in Section 3.2 and 3.3 respectively. Then, in Section 3.4 we introduce different strategies of sharing image backbones for hand and object pose predictors as well as the SDF feature encoder. Finally, the training strategy of our model is described in Section 3.5.
3.2. Kinematic Feature Encoding

**Hand and object pose estimation.** Directly regressing hand pose parameters of MANO from image features [11,19,21] has proved to be difficult [6,28,66]. In contrast, predicting sparse 3D joint locations is easier and can achieve higher accuracy. Therefore, we first train a 3D hand joint prediction model which produces volumetric heatmaps [38,44] for 21 hand joints. We use a differentiable soft-argmax operator [50] to extract 3D coordinates $\psi_h \in \mathbb{R}^{21 \times 3}$ of hand joints from the heatmaps. We then obtain an analytic solution for hand poses $\theta_h \in \mathbb{R}^{16 \times 3}$, $\phi_h \in \mathbb{R}^{16 \times 3}$ from estimated 3D joints $\psi_h$ using inverse kinematics, where each $\theta_{h,i} \in \mathbb{R}^3$ and $\theta_{h,i} \in \mathbb{R}^3$ denote the relative pose of the $i_{th}$ joint in terms of rotation and translation with respect to its ancestor joint. Here, we only calculate the rotation and use the default limb lengths provided by the MANO model. Specifically, we first compute the pose of the hand wrist using the template pose defined in MANO, and then follow the hand kinematic chain to solve the pose of other finger joints recursively. More details are presented in the supplementary material.

For the object pose estimation, it is often difficult to accurately estimate the rotation of the object since many objects have a high degree of symmetry and are often occluded by hands. We therefore follow [11] and only estimate the center position of the object $\psi_o \in \mathbb{R}^3$ relative to the hand wrist.

**Hand kinematic feature.** Given the 3D point $x$, we generate the hand kinematic feature $e_h \in \mathbb{R}^{51}$ by transforming $x$ into canonical coordinate frames defined by hand joints. Figure 3(top,right) illustrates the proposed geometry transformation for the hand. For the $i_{th}$ hand joint pose $\theta_{h,i}, \phi_{h,i}$, the pose transformation $T_i(x, \theta_{h,i}, \phi_{h,i})$ to obtain the local hand kinematic feature $e_{h,i} \in \mathbb{R}^3$ is defined as

$$e_{h,i} = T_p(x, \theta_{h,i}, \phi_{h,i}) = \tilde{H}(G_{h,i}^{-1} \cdot H(x)),$$

where $A(i)$ denotes the ordered set of ancestors of the $i_{th}$ joint. We use Rodrigues formula $\exp(\cdot)$ to convert $\theta_{h,i}$ into the form of a rotation matrix. By traversing the hand kinematic chain, we obtain the global transformation $G_{h,i} \in \mathbb{R}^{4 \times 4}$ for the $i_{th}$ joint. Then, we take the inverse of $G_{h,i}$ to transform $x$ into the $i_{th}$ hand joint canonical coordinates. $H(\cdot)$ transforms $x$ into homogeneous coordinates while $\tilde{H}(\cdot)$ transforms homogeneous coordinates back to Euclidean coordinates. Given local kinematic features $e_{h,i}$, the hand kinematic feature $e_h \in \mathbb{R}^{51}$ is defined as

$$e_h = [x, e_{h,1}, \cdots, e_{h,16}].$$

**Object kinematic feature.** To obtain geometry-aware SDF for object reconstruction, we propose object kinematic feature $e_o \in \mathbb{R}^{72}$. Following [11], we use estimated object center $\psi_o$ to transform $x$ into the object canonical coordinate frame by the translation transformation $x_{oc} = T_t(x, \psi_o) = x - \psi_o$. As the grasping hand pose also gives hints about the shape of the manipulated object, similar to [63] we incorporate the knowledge of hand poses into object reconstruction. To this end, for each joint $i$ and its estimated 3D location $\psi_{h,i}$, we transform $x$ by translation as

$$e_{o,i} = T_t(x, \psi_{h,i}) = x - \psi_{j,i}.$$  

Figure 3(bottom,right) illustrates the proposed geometry transformation for the object kinematic feature.

3.3. Visual Feature Encoding

**Geometry-aligned visual feature.** Previous works [11,26] typically predict signed distances from global image features that lack spatial resolution. Motivated by [47], we aim to generate geometry-aligned local image features for each input point $x$. Assume $\psi_i^* \in \mathbb{R}^{16 \times 16 \times d}$ is the feature map generated from the SDF feature encoder, e.g. a ResNet model [22], where $16 \times 16$ is the spatial feature resolution and $d$ is the feature dimension. We project the 3D input point $x$ to $\hat{x}$
on the image plane with the camera projection matrix and use bilinear sampling to obtain a local feature \( e_v \) from the location on the feature map corresponding to \( \hat{x} \).

**Temporally-enhanced visual feature.** To improve the robustness of visual features in a single frame \( I_t \) from occlusion or motion blur, we propose to exploit temporal information from videos to refine \( v_t \). Note that due to non-rigid hand motions, we do not assume video frames to contain different views of the same rigid scene. We make use of the spatial-temporal transformer architecture [1, 4] to efficiently propagate image features across frames. Assume \( v_{t-1} \) \( v_{t+1} \) are the feature maps from neighboring frames of \( I_t \) in a video. We flatten all the feature maps as a sequence in the spatial-temporal dimension leading to \( 3 \times 16 \times 16 \) tokens fed into the transformer model. We reshape the output features of the transformer into a feature map again for \( I_t \), denoted as \( v_t \in \mathbb{R}^{16 \times 16 \times d} \). By aggregating spatial and temporal information from multiple frames, \( v_t \) becomes more robust to the noise and can potentially produce more stable reconstruction results compared to \( v_t \). Our full gSDF model relies on the feature map \( v_t \) to compute the local visual feature \( e_v \) for the given input point \( x \).

### 3.4. Image Backbone Sharing Strategy

As shown in Figure 2, our model contains three branches for hand and object pose estimations as well as for SDF feature encoding. These different branches may share image backbones which might be beneficial with the multi-task learning. In this section, we describe three alternative strategies for sharing image backbones in our model.

**Single image backbone (Figure 4a).** We only employ one single image backbone for both pose and shape predictions. This is the strategy used in AlignSDF [11].

**Symmetric image backbone (Figure 4b).** To disentangle pose and shape learning, we share the image backbone for hand and object pose estimation, but use a different backbone to extract visual features for SDFs learning.

**Asymmetric image backbone (Figure 4c).** Since hand pose estimation plays a critical role in the task, we use a separate backbone to predict the hand pose, while share the image backbone for object pose predictor and SDF feature encoder.

### 3.5. Training

We apply a two-stage training strategy. In the first stage, we train the hand pose predictor to predict hand joint coordinates \( \psi_h \) with \( \ell_2 \) loss \( L_{hp} \) and an ordinal loss [43] \( L_{ord} \) to penalize the case if the predicted depth order between the \( i_{th} \) joint and the \( j_{th} \) joint is misaligned with the ground-truth relation \( I_{ord}^{h,i,j} \), which are:

\[
L_{hp} = \frac{1}{21} \sum_{i=1}^{21} \left\| \psi_{h,i} - \hat{\psi}_{h,i} \right\|_2^2, \tag{6}
\]

\[
L_{ord} = \sum_{j=2}^{21} \sum_{i=1}^{j-1} I^{ord}_{i,j} \times \left| \left( \psi_{h,i} - \psi_{h,j} \right) \cdot \vec{n} \right|. \tag{7}
\]

where \( \vec{n} \in \mathbb{R}^3 \) denotes the viewpoint direction. We randomly sample twenty virtual views to optimize \( L_{ord} \). Since the proposed kinematic features are based on the predicted hand joints \( \psi_h \), we empirically find that pretraining the hand joint predictor in the first stage and then freezing its weights can achieve better performance.

In the second training stage, we learn all the modules except the hand joint predictor in an end-to-end manner. We use the \( \ell_2 \) loss \( L_{op} \) to predict the object pose \( \psi_o \) as follows:

\[
L_{op} = \left\| \psi_o - \hat{\psi}_o \right\|_2^2 \tag{8}
\]

where \( \hat{\psi}_o \) denote the ground-truth location for the object center. To train the SDFs, we sample many 3D points around the hand-object surface and calculate their ground-truth signed distances to the hand mesh and the object mesh. We use \( \ell_1 \) loss to optimize the SDF decoders:

\[
L_{hsdf} = \left\| \text{SDF}_{hand} - \tilde{\text{SDF}}_{hand} \right\|_1, \tag{9}
\]

\[
L_{osdf} = \left\| \text{SDF}_{obj} - \tilde{\text{SDF}}_{obj} \right\|_1,
\]

where \( \text{SDF}_{hand} \) and \( \tilde{\text{SDF}}_{obj} \) denote ground-truth signed distances to the hand and the object, respectively. The overall training objective \( L_{shape} \) in the second training stage is:

\[
L_{shape} = L_{op} + 0.5 \times L_{hsdf} + 0.5 \times L_{osdf}. \tag{10}
\]
4. Experiments

We conduct extensive experiments on two 3D hand-object reconstruction benchmarks to evaluate the effectiveness of our proposed gSDF model.

4.1. Datasets

ObMan [21] is a large-scale synthetic dataset that contains diverse hand grasping poses on a wide range of objects imported from ShapeNet [8]. We follow previous methods [11, 26, 42, 63] to generate data for SDFs training. First, we remove meshes that contain too many double-sided triangles, which results in 87,190 hand-object meshes. Then, we fit the hand-object mesh into a unit cube and sample 40,000 points inside the cube. For each sampled point, we compute its signed distance to the ground-truth hand mesh and object mesh, respectively. At test time, we report the performance on the whole ObMan test set of 6,285 testing samples.

DexYCB [9] is currently the largest real dataset that captures hand and object interactions in videos. Following [11, 60], we focus on right-hand samples and use the official s0 split. We follow the same steps as in ObMan to obtain SDF training samples. To reduce the temporal redundancy, we downsample the video data to 6 frames per second, which results in 29,656 training samples and 5,928 testing samples.

4.2. Evaluation metrics

We follow prior works to comprehensively evaluate the 3D reconstructions with multiple metrics as below.

**Hand Chamfer Distance (CD_h).** We evaluate Chamfer distance (cm²) between our reconstructed hand mesh and the ground-truth hand mesh. We follow previous works [11, 26] to optimize the scale and translation to align the reconstructed mesh with the ground truth and sample 30,000 points on both meshes to compute Chamfer distance. We report the median Chamfer distance on the test set to reflect the quality of our reconstructed hand mesh.

**Hand F-score (FS_h).** Since Chamfer distance is vulnerable to outliers [52, 63], we also report the F-score to evaluate the predicted hand mesh. After aligning the hand mesh with its ground truth, we report F-score at 1mm (FS_h@1) and 5mm (FS_h@5) thresholds.

**Object Chamfer Distance (CD_o).** Following [11, 26], we first use the optimized hand scale and translation to transform the reconstructed object mesh. Then, we follow the same process as CD_h to compute CD_o (cm²) and evaluate the quality of our reconstructed object mesh.

**Object F-score (FS_o).** We follow the previous work [63] to evaluate the reconstructed object mesh using F-score at 5 mm (FS_o@5) and 10 mm (FS_o@10) thresholds.

**Hand Joint Error (E_h).** To measure the hand pose estimation accuracy, we compute the mean joint error (cm) relative to the hand wrist over all 21 joints in the form of ℓ₂ distance.

**Object Center Error (E_o).** To evaluate the accuracy of our predicted object translation, we report the ℓ₂ distance (cm) between the prediction and its ground truth.

Additionally, we report Contact ratio (Cr), Penetration depth (Pd) and Intersection volume (Iv) [11, 21, 26, 60, 62] to present more details about the interaction between the hand mesh and the object mesh. Please see supplementary material for more details.

4.3. Implementation details

**Model architecture.** We use ResNet-18 [22] as our image backbone. For hand and object pose estimation, we adopt volumetric heatmaps of spatial resolution 64 × 64 × 64 to localize hand joints and the object center in 3D space. For the spatial-temporal transformer, we use 16 transformer layers with 4 attention heads. We present more details about our model architecture in supplementary material.

**Training details.** We take the image crop of the hand-object region according to their bounding boxes for DexYCB benchmark. Then, we modify camera intrinsic and extrinsic parameters [35, 64] accordingly and take the cropped image as the input to our model. The spatial size of input images is 256 × 256 for all our models. We perform data augmentation including rotation ([-45°, 45°]) and color jittering. During SDF training, we randomly sample 1000 points (500 points inside the mesh and 500 points outside the mesh) for the hand and the object, respectively. We train our model with a batch size of 256 for 1600 epochs on both ObMan and DexYCB using the Adam optimizer [27] with 4 NVIDIA RTX 3090 GPUs. We use an initial learning rate of 1 × 10⁻⁴ and decay it by half every 600 epochs. It takes 22 hours for training on DexYCB and 60 hours on ObMan dataset.

4.4. Ablation studies

We carry out ablations on the DexYCB dataset to validate different components in our gSDF model. We evaluate different settings of hand kinematic features (K_h in Table 1), object kinematic features (K_o in Table 2), and visual features.
Table 3. Hand-object reconstruction performance with different visual features on DexYCB dataset. The visual features are combined with the best kinematic features $K_h^3$ (Table 1) and $K_o^3$ (Table 2) to reconstruct hand and object respectively.

<table>
<thead>
<tr>
<th>Transformer</th>
<th>Global</th>
<th>Local</th>
<th>Spatial</th>
<th>Temp.</th>
<th>$CD_h$</th>
<th>$FS_h@1$</th>
<th>$FS_h@5$</th>
<th>$CD_o$</th>
<th>$FS_o@5$</th>
<th>$FS_o@10$</th>
<th>$E_h$</th>
<th>$E_o$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$V_1$</td>
<td>✓</td>
<td>×</td>
<td>×</td>
<td>×</td>
<td>0.317</td>
<td>0.171</td>
<td>0.788</td>
<td>1.71</td>
<td>0.418</td>
<td>0.689</td>
<td>1.44</td>
<td>1.91</td>
</tr>
<tr>
<td>$V_2$</td>
<td>×</td>
<td>✓</td>
<td>✓</td>
<td>×</td>
<td>0.310</td>
<td>0.172</td>
<td>0.795</td>
<td>1.71</td>
<td>0.426</td>
<td>0.694</td>
<td>1.44</td>
<td>1.98</td>
</tr>
<tr>
<td>$V_3$</td>
<td>×</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>0.304</td>
<td>0.174</td>
<td>0.797</td>
<td>1.60</td>
<td>0.434</td>
<td>0.703</td>
<td>1.44</td>
<td>1.94</td>
</tr>
<tr>
<td>$V_4$</td>
<td>×</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>0.302</td>
<td>0.177</td>
<td>0.801</td>
<td>1.55</td>
<td>0.437</td>
<td>0.709</td>
<td>1.44</td>
<td>1.96</td>
</tr>
</tbody>
</table>

Table 4. Hand-object reconstruction performance using different image backbone sharing strategies on DexYCB dataset. The ablation is carried out with visual features $V_i$ and kinematic features $K_h^3$ and $K_o^3$.

<table>
<thead>
<tr>
<th>Backbone</th>
<th>$CD_h$</th>
<th>$FS_h@1$</th>
<th>$FS_h@5$</th>
<th>$CD_o$</th>
<th>$FS_o@5$</th>
<th>$FS_o@10$</th>
<th>$E_h$</th>
<th>$E_o$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Single</td>
<td>0.411</td>
<td>0.148</td>
<td>0.741</td>
<td>1.88</td>
<td>0.402</td>
<td>0.674</td>
<td>1.72</td>
<td>1.83</td>
</tr>
<tr>
<td>Symmetric</td>
<td>0.324</td>
<td>0.168</td>
<td>0.779</td>
<td>1.84</td>
<td>0.405</td>
<td>0.672</td>
<td>1.46</td>
<td>1.93</td>
</tr>
<tr>
<td>Asymmetric</td>
<td>0.317</td>
<td>0.171</td>
<td>0.788</td>
<td>1.71</td>
<td>0.418</td>
<td>0.689</td>
<td>1.44</td>
<td>1.91</td>
</tr>
</tbody>
</table>

(V_ in Table 3). We use the asymmetric image backbone if not otherwise mentioned.

**Hand kinematic feature.** In Table 1, we evaluate the contribution of the proposed hand kinematic features for 3D hand reconstruction. The model in $K_h^3$ does not use any pose priors to transform the 3D point. The model in $K_h^3$ only uses the hand wrist pose to transform the 3D point as AlignSDF [11]. Our model in $K_h^3$ computes the transformations to all the hand joints, which achieves the best performance on all the evaluation metrics. Compared to $K_h^3$ without any pose priors, our model achieves more than 12% and 9% improvement on $CD_h$ and $FS_h@1$, respectively. Compared to $K_h^3$ with only hand wrist, our model greatly reduces the hand Chamfer distance from 0.344 cm² to 0.317 cm², leading to 7.8% relative gains. These results demonstrate the significance of pose priors and the advantage of gSDF for 3D hand reconstruction.

**Object kinematic feature.** In Table 2, we validate the effectiveness of our proposed object kinematic feature. The model in $K_o^3$ does not contain any pose priors, while the model in $K_o^2$ aligns query points to the object center as in [11]. Our model in $K_o^3$ further employs the hand pose to produce the object kinematic feature, which significantly boosts the performance for the object reconstruction on different metrics. Compared to $K_o^2$, our proposed object kinematic feature achieves more than 11% and 5.5% improvement on $CD_o$ and $FS_o@5$, respectively.

**Visual features.** We compare different visual features for SDF prediction in Table 3. $V_1$ uses the global visual feature e.g., the average pooling of ResNet feature map as in previous works [11, 26]. Our local visual features $V_2$ derived from the geometry alignment with the query point reduces the hand Chamfer distance from 0.317 cm² to 0.310 cm². However, it shows less improvement on the object shape accuracy. In $V_3$ and $V_4$, we use the transformer model to refine the feature maps. To ablate the improvement from the transformer architecture and from the temporal information in videos, we only use transformer for each single frame in $V_3$ while use it for multiple frames in $V_4$. We can see that the transformer architecture alone is beneficial for the reconstruction. Enhancing the visual features with temporal contexts further improves the performance in terms of all the evaluation metrics especially for the objects. In Figure 5, compared with our single-frame model built with the transformer, our video model can make more robust predictions under some hard cases (e.g., motion blur). Although the reconstruction of the can is not accurate in the first example, our model tends to produces more regular shapes.

**Image backbone sharing strategy.** Results of using different strategies for image backbone sharing are presented in Table 4. We train all the three models using the two-stage strategy described in Section 3.5. The model with one single backbone achieves the worst performance under most of the evaluation metrics. This is because the pose learning and shape learning compete with each other during training. The symmetric strategy to separate backbones for pose and SDFs performs better than the single backbone model. Our asymmetric strategy with a separate backbone for hand pose estimation and a shared backbone for object pose and SDF feature encoder achieves the best performance. We also em-

![Figure 5. The qualitative comparison between our single-frame model built with the transformer and our video model.](image-url)
Table 5. Comparison with state-of-the-art methods on the image ObMan dataset.

<table>
<thead>
<tr>
<th>Methods</th>
<th>$CD_h$ $\downarrow$</th>
<th>$FS_h@1$ $\uparrow$</th>
<th>$FS_h@5$ $\uparrow$</th>
<th>$CD_o$ $\downarrow$</th>
<th>$FS_o@5$ $\uparrow$</th>
<th>$FS_o@10$ $\uparrow$</th>
<th>$E_h$ $\downarrow$</th>
<th>$E_o$ $\downarrow$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hasson et al. [21]</td>
<td>0.415</td>
<td>0.138</td>
<td>0.751</td>
<td>3.60</td>
<td>0.359</td>
<td>0.590</td>
<td>1.13</td>
<td>-</td>
</tr>
<tr>
<td>Karunratanakul et al. [26]</td>
<td>0.261</td>
<td>-</td>
<td>-</td>
<td>6.80</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Ye et al. [63]</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>0.420</td>
<td>0.630</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Chen et al. [11]</td>
<td>0.136</td>
<td>0.302</td>
<td>0.913</td>
<td>3.38</td>
<td>0.404</td>
<td>0.636</td>
<td>1.27</td>
<td>3.29</td>
</tr>
<tr>
<td><strong>gSDF (Ours)</strong></td>
<td><strong>0.112</strong></td>
<td><strong>0.332</strong></td>
<td><strong>0.935</strong></td>
<td><strong>3.14</strong></td>
<td><strong>0.438</strong></td>
<td><strong>0.660</strong></td>
<td><strong>0.93</strong></td>
<td><strong>3.43</strong></td>
</tr>
</tbody>
</table>

Table 6. Comparison with state-of-the-art methods on the video DexYCB dataset.

<table>
<thead>
<tr>
<th>Methods</th>
<th>$CD_h$ $\downarrow$</th>
<th>$FS_h@1$ $\uparrow$</th>
<th>$FS_h@5$ $\uparrow$</th>
<th>$CD_o$ $\downarrow$</th>
<th>$FS_o@5$ $\uparrow$</th>
<th>$FS_o@10$ $\uparrow$</th>
<th>$E_h$ $\downarrow$</th>
<th>$E_o$ $\downarrow$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hasson et al. [21]</td>
<td>0.537</td>
<td>0.115</td>
<td>0.647</td>
<td>1.94</td>
<td>0.383</td>
<td>0.642</td>
<td>1.67</td>
<td>-</td>
</tr>
<tr>
<td>Karunratanakul et al. [26]</td>
<td>0.364</td>
<td>0.154</td>
<td>0.764</td>
<td>2.06</td>
<td>0.392</td>
<td>0.660</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Chen et al. [11]</td>
<td>0.358</td>
<td>0.162</td>
<td>0.767</td>
<td>1.83</td>
<td>0.410</td>
<td>0.679</td>
<td>1.58</td>
<td><strong>1.78</strong></td>
</tr>
<tr>
<td>Chen et al. [11]†</td>
<td>0.344</td>
<td>0.167</td>
<td>0.776</td>
<td>1.81</td>
<td>0.413</td>
<td>0.687</td>
<td>1.57</td>
<td>1.93</td>
</tr>
<tr>
<td><strong>gSDF (Ours)</strong></td>
<td><strong>0.302</strong></td>
<td><strong>0.177</strong></td>
<td><strong>0.801</strong></td>
<td><strong>1.55</strong></td>
<td><strong>0.437</strong></td>
<td><strong>0.709</strong></td>
<td><strong>1.44</strong></td>
<td><strong>1.96</strong></td>
</tr>
</tbody>
</table>

Figure 6. Qualitative results of our model on test images from the ObMan and DexYCB benchmarks. Our model produces convincing results for different grasping poses and diverse objects.

5. Conclusion

In this work, we propose a geometry-driven SDF (gSDF) approach for 3D hand and object reconstruction. We explicitly model the underlying 3D geometry to guide the SDF learning. We first estimate poses of hands and objects according to kinematic chains of pose transformations, and then derive kinematic features and local visual features using the geometry information for signed distance prediction. Extensive experiments on ObMan and DexYCB datasets demonstrate the effectiveness of our proposed method.

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. We thank Yana Hasson for helpful discussions.
References


[40] Franziska Mueller, Micah Davis, Florian Bernard, Oleksandr Sotynchenko, Mickeal Verschoor, Miguel A Otaduy, Dan Casas, and Christian Theobalt. Real-time pose and shape reconstruction of two interacting hands with a single depth camera. TOG, 2019. 2

[41] Iason Oikonomidis, Nikolaos Kyriazis, and Antonis A Argyros. Full DOF tracking of a interacting hand with an object by modeling occlusions and physical constraints. In ICCV, 2011. 2


[48] Xiao Sun, Bin Xiao, Fangyin Wei, Shuang Liang, and Yichen Wei. Integral human pose regression. In ECCV, 2018. 2, 4


[54] Dimitrios Tzionas and Juergen Gall. 3D object reconstruction from hand-object interactions. In ICCV, 2015. 2


[56] Yangang Wang, Jianyuan Min, Jianjie Zhang, Yebin Liu, Feng Xu, Qionghai Dai, and Jinxiang Chai. Video-based hand manipulation capture through composite motion control. TOG, 2013. 2


[58] Lixin Yang, Kailin Li, Xinyu Zhan, Jun Lv, Wenchiang Xu, Jiefeng Li, and Cewu Lu. AntiBoost: Boosting articulated 3D hand-object pose estimation via online exploration and synthesis. In CVPR, 2022. 2, 6

[59] Lixin Yang, Kailin Li, Xinyu Zhan, Fei Wu, Anran Xu, Liu Liu, and Cewu Lu. OakInk: A large-scale knowledge repository for understanding hand-object interaction. In CVPR, 2022. 1, 2

[60] Lixin Yang, Xinyu Zhan, Kailin Li, Wenchiang Xu, Jiefeng Li, and Cewu Lu. CPUF: Learning a contact potential field to model the hand-object interaction. In ICCV, 2021. 1, 2, 6

[61] Yufei Ye, Abhinav Gupta, and Shubham Tulsiani. What’s in your hands? 3D reconstruction of generic objects in hands. In CVPR, 2022. 2, 3, 4, 6, 8


[63] Shanzin Yuan, Guillermo Garcia-Hernando, Björn Stenger, Gyeongsik Moon, Ju Yong Chang, Kyoung Mu Lee, Pavlo Molchanov, Jan Kautz, Sina Honari, Liuhao Ge, Junsong Yuan, Xinghao Chen, Gjiunmin Wang, Fan Yang, Kai Akiyama, Yang Wu, Qingfu Wan, Meysam Madadi, Sergio Escalera, Shile Li, Dongheui Lee, Jason Oikonomidis, Antonis Argyros, and Tae-Kyun Kim. Depth-based 3D hand pose estimation:
From current achievements to future goals. In CVPR, June 2018.
