Learning Explicit Contact for Implicit Reconstruction of Hand-held Objects from Monocular Images

Junxing Hu[1,2], Hongwen Zhang[3], Zerui Chen[4], Mengcheng Li[3]
Yunlong Wang[2], Yebin Liu[3], Zhenan Sun[2]
[1]University of Chinese Academy of Sciences, [2]CASIA, [3]Tsinghua University, [4]Inria
https://junxinghu.github.io/projects/hoi.html

Abstract

Reconstructing hand-held objects from monocular RGB images is an appealing yet challenging task. In this task, contacts between hands and objects provide important cues for recovering the 3D geometry of the hand-held objects. Though recent works have employed implicit functions to achieve impressive progress, they ignore formulating contacts in their frameworks, which results in producing less realistic object meshes. In this work, we explore how to model contacts in an explicit way to benefit the implicit reconstruction of hand-held objects. Our method consists of two components: *explicit contact prediction* and *implicit shape reconstruction*. In the first part, we propose a new subtask of directly estimating 3D hand-object contacts from a single image. The part-level and vertex-level graph-based transformers are cascaded and jointly learned in a coarse-to-fine manner for more accurate contact probabilities. In the second part, we introduce a novel method to diffuse estimated contact states from the hand mesh surface to nearby 3D space and leverage diffused contact probabilities to construct the implicit neural representation for the manipulated object. Benefiting from estimating the interaction patterns between the hand and the object, our method can reconstruct more realistic object meshes, especially for object parts that are in contact with hands. Extensive experiments on challenging benchmarks show that the proposed method outperforms the current state of the arts by a great margin.

1 Introduction

Reconstructing hand-held objects from monocular images is essential to understand the interactions between humans and the physical world. Toward this goal, recent progress has been achieved in the individual reconstruction of hands [1, 3, 7, 19, 28, 45], objects [10, 16, 32, 36, 39, 50], and their joint reconstruction [8, 9, 20, 21, 25, 54, 55]. However, this task remains very challenging due to the complexity of hand poses and the diversity of interacting objects.

As hand-held objects involve the grasp configuration between hands and objects, the contacts play essential roles in modeling hand-object interactions. To improve the interaction, there have been several attempts to model the contact in different representations, including using contacts to optimize meshes [21], the contact potential field [54], or the grasping field [25]. However, these methods only model contacts as an additional loss function, which miss the chance to construct and exploit contact priors to simplify the 3D reconstruction problem.

Our key observation is that the contacts between hands and objects provide important cues for recovering the 3D geometry of the hand-held object. Modeling contacts between hands and objects can compensate for the lack of 3D information in monocular RGB images and makes it easier to infer the shape of the hand-held objects, especially for parts that are in contact with hands. Though previous methods have included contact losses [21, 25] or optimization objectives [14, 54] in their

Preprint. Under review.

reconstruction pipelines, they do not consider the usage of contacts as an intermediate representation to benefit the 3D reconstruction. In this work, we explore how to construct contact priors from the monocular RGB image to help recover the 3D object geometry. Specifically, we first predict contact points explicitly on the hand mesh surface. To our knowledge, estimating contact states from a single RGB image is explored only for human body mesh [12, 13, 23] without focusing on the hand. To this end, we introduce a novel coarse-to-fine learning framework to jointly learn part-level and vertex-level contact states. In addition, we utilize the graph-based transformer which combines graph convolutions with transformers to better accumulate relevant features among adjacent nodes in the hand mesh and obtain more robust contact predictions.

Then, we attempt to exploit predicted contact states to simplify the 3D reconstruction task. Here, we follow the previous work [55] to model hand-held objects with deep implicit functions [36], which can generate realistic and high-resolution object meshes. However, how to make implicit functions take good advantage of estimated contact states is also challenging and remains unsolved. The main challenge is that contact points are distributed on the hand surface in the discrete form, while implicit functions have continuous values in the whole 3D volume. To tackle the difficulty, we employ sparse convolutions to diffuse these discrete contact states from the hand surface to the 3D space. Then, the implicit function can naturally query corresponding contact features for a given 3D point and improve the neural implicit reconstruction. We conduct extensive experiments on HO3D [17] and OakInk [53] benchmarks to show that our method can reconstruct high-quality object meshes that interact faithfully with hands.

To sum up, the main contributions of this work can be listed as follows:

- We propose to leverage contact priors for better reconstruction of hand-held objects. To estimate contact states more accurately, we introduce a novel framework that jointly optimizes part-level and vertex-level contact states in a coarse-to-fine manner.
- To make discrete contact states compatible with continuous implicit shape functions, we propose to diffuse contact features from the hand mesh surface to the whole 3D volume, which enables the continuous query of contact features for implicit object reconstruction.
- We conduct extensive experiments on HO3D and OakInk benchmarks to validate the effectiveness of our method. Our method can produce more realistic hand-held object meshes and advance state-of-the-art accuracy.

2 Related Work

Our work focuses on reconstructing hand-held objects from monocular RGB images. In this section, we first review related works in the field of 3D hand-object reconstruction. Then, we discuss how to leverage contact information to improve the quality of 3D reconstruction.

3D Hand-object Reconstruction. This task aims to reconstruct the 3D geometry of hands and hand-held objects from images. Existing approaches can be generally classified into two categories: multi-view and single-view methods. Multi-view methods [6, 17, 35, 51, 53] employ multiple cameras positioned at different viewpoints to infer the 3D structure of the grasping scenario. Though this type of method can generate very accurate 3D reconstruction results, they need careful camera calibrations and are inconvenient to deploy in the wild scene. Single-view methods only need monocular sensors [8, 9, 20–22, 25, 29, 47, 54, 55, 57, 58] as inputs and are flexible to apply in real practice. In this work, we use the most common monocular RGB images as inputs. However, given the ill-posed nature, it is quite challenging to infer the 3D structure only from monocular RGB cues. To alleviate the difficulty of the hand reconstruction problem, Hasson *et al.* [20, 21] propose to employ the parametric hand model MANO [45], which encodes rich hand priors, to predict the hand mesh. To produce more realistic hand meshes, recent works [8, 9, 25] employ the neural implicit function [36] to model the hand shape and use estimated hand pose priors [8, 9] to simplify the hand shape learning. However, compared with the hand part, hand-held object reconstruction is even more challenging. Since there are thousands of manipulated objects in our daily lives, it is difficult to make a unified object mesh template like MANO or estimate 6D poses reliably for diverse objects, especially for symmetric objects. Given its difficulty, some existing works [20, 52, 54] even make a strong assumption that the perfect object model is known at test time and only predicts its 6D pose. A recent work [55] relaxes this assumption and proposes to leverage estimated hand poses to benefit the implicit reconstruction of hand-held objects. In this work, we go a step further and argue that contacts

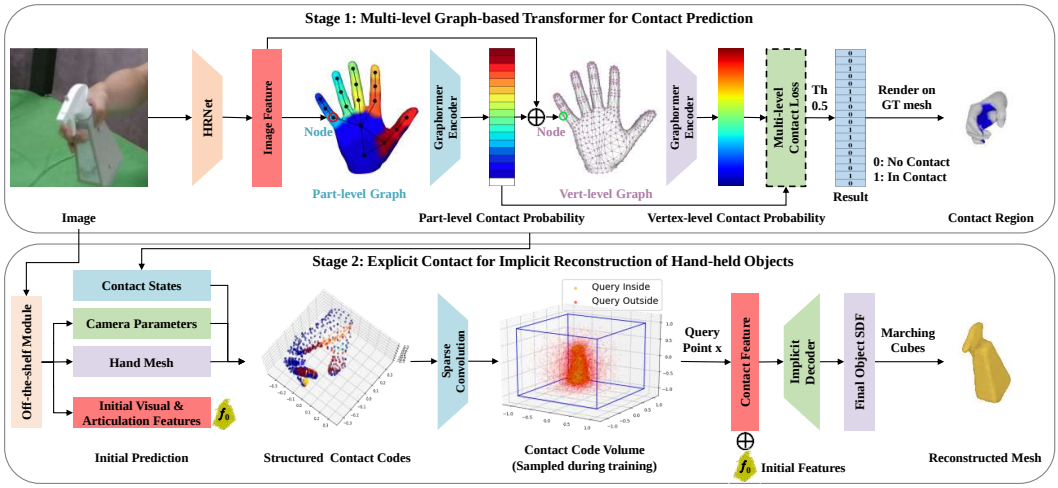

Figure 1: The overview of learning explicit contact for implicit reconstruction. In the first stage, the method estimates contact regions of the hand given a monocular RGB image. Based on the template MANO mesh, part- and vertex-level graph-based transformers are constructed and cascaded together for accurate predictions. In the second stage, the estimated contact is used to construct the implicit neural representation. An off-the-shelf module is first utilized to produce the camera parameters, hand mesh, and initial features from the image. Then, the structured contact codes are generated by anchoring contact probabilities to the hand mesh surface. After sparse convolutions, the contact states on the hand surface are diffused to its nearby 3D space, which facilitates the perception and reconstruction of the manipulated object.

between hands and objects could provide important cues for 3D reconstruction and introduce a novel framework to generate more realistic object meshes that interact with hands.

Contacts in Object Reconstruction and Manipulation. Learning to model and reconstruct the 3D geometry of objects from monocular images has been a crown jewel in the field of computer vision [34, 44]. Previous works usually represent the 3D object using explicit representations (*e.g.*, meshes [16, 50], point clouds [41, 42] or voxels [11, 38, 43]) and use deep neural networks to predict them. In recent years, neural implicit functions [10, 32, 36] have gradually become a popular paradigm for 3D reconstruction. It is seamlessly compatible with neural networks and can theoretically reconstruct objects at unlimited resolution. However, the implicit function itself does not contain object surface priors, which makes it hard to fit diverse object surfaces. In this work, we construct a surface prior using contacts between hands and objects and simplify the learning problem. Actually, some works in robotics [2, 4, 30, 56] have shown that contacts could provide rich cues about the object shape and how to manipulate the given object. Some recent systems[24, 56] can successfully manipulate different objects by using contact sensors. However, how to use contacts to benefit hand-held reconstruction is under-explored in our task. Previous works only use contact information in an implicit way. Methods using explicit object representations[14, 21, 54] introduce contact loss terms to encourage objects to be close to reconstructed hand meshes. Some recent efforts [25, 55] also introduce contact loss terms in the context of neural implicit representation. Different from them, we model and predict contact states explicitly and successfully leverage constructed contact priors to improve the quality of neural implicit reconstruction.

3 Method

In this section, we describe the technical details of the proposed method. As shown in Fig. 1, our method consists of two stages: explicit contact prediction and implicit shape reconstruction. In the first stage (Section 3.1), we propose to predict part-level and vertex-level hand contact states in a coarse-to-fine manner. A graph-based transformer model is introduced to estimate contact probabilities more accurately. In the second stage (Section 3.2), we present a novel method to leverage estimated contact states to improve the neural implicit reconstruction of hand-held objects.

3.1 Explicit Contact Prediction

Given a single RGB image I, our method first predicts the contact regions between the hands and objects. Specifically, we estimate contact probabilities within $[0, 1]$ on hand meshes to measure the likelihood of the region touching the object. In our method, the contact probabilities are predicted from coarse to fine and denoted as $\boldsymbol{C_p} = \{\boldsymbol{c_p}^i \in [0,1]\}_{i=1}^{N_p}$ and $\boldsymbol{C_v} = \{\boldsymbol{c_v}^i \in [0,1]\}_{i=1}^{N_v}$ for the part-level and vertex-level contacts, where N_p and N_v are the number of the hand parts and hand mesh vertices, respectively.

Multi-level Contact Graphs. For more accurate predictions of the contact probabilities, multi-level contact graphs are leveraged to process the surface regions in the part and vertex levels such that the contact can be jointly learned from coarse to fine. In this way, the contact graphs are built under different granularities and learn the hand-object interactions in various scales.

Considering that the hand mesh can be naturally represented as a graph, we build the contact graphs based on the template MANO mesh [45]. Specifically, the part-level graph G_p with N_p nodes is generated relying on a coarse division of the hand regions. According to statistical contact frequency, the hand surface is divided into N_p subregions, including $(N_p - 1)$ subregions on the hand palm and one subregion on the back side of the hand. When building graph G_p, the center point of each part of the MANO template is taken as a graph node. For each graph node, its features are the concatenation of the image-based feature and its 3D coordinates. As shown in the first stage in Fig. 1, an image feature $f \in \mathbb{R}^D$ with the length of D is extracted from I by using an HRNet backbone [49]. Therefore, each part-level graph node feature of G_p is $\boldsymbol{g_p^i} \in \mathbb{R}^{D+3}, i = \{1, 2, \ldots, N_p\}$ and the adjacency matrix is encoded as the physical contact relationship between nodes. On the other hand, the vertex-level graph G_v is generated based on the N_v mesh vertices with an adjacency matrix from the MANO template. In addition to the image feature, the vertex-level node features of G_v also include the part-level contact probability $\boldsymbol{C_p}$, resulting in the node feature $\boldsymbol{g_v^i} \in \mathbb{R}^{D+N_p+3}, i = \{1, 2, \ldots, N_v\}$.

Graph-based Transformer for Contact Prediction. In hand-object interaction, the contact area is usually occluded by hands or objects, which requires the network to perceive local details and global information. Following Graphormer [31], our contact estimators are designed as graph-based transformers that incorporate the graph convolution [27] into the transformer block [48]. In this way, the graph convolution focuses on fine-grained local interactions, while the latter encodes the global relationships of the whole hand regions. As the contacts are predicted at the part and vertex levels, the architectures of the coarse and fine contact estimators are also built upon the graphs G_p and G_v, respectively. Specifically, the coarse and fine contact estimators have N_p and N_v input tokens, which correspond to the same number of nodes in the graphs. Moreover, the two contact estimators have different hidden sizes in their transformer blocks. In practice, we find that a hidden size of 256 is sufficient for the part-level contact estimation, and the three blocks with hidden sizes of 1024, 256, and 64 work well for the vertex-level contact estimator.

For both the two contact estimators, the size of the output token is set to one. Similar to the settings in BSTRO [23], a sigmoid function is used to convert output tokens to contact probabilities in the range of [0, 1], and we extract contact points with probabilities greater than 0.5.

3.2 Explicit Contact for Implicit Object Reconstruction

As shown in the second stage in Fig. 1, given the explicit contact prediction $\boldsymbol{C_p}$ and $\boldsymbol{C_v}$ with the hand mesh, our method first builds structured contact codes in a normalized 3D space. Then, they are fed into a sparse convolutional network to generate the contact code volumes $\boldsymbol{V}$ at different resolutions. This operation diffuses the contact states on the hand surface to the nearby 3D space and can be sampled continuously as additional conditions for the implicit reconstruction of objects.

Initial Prediction. Given an RGB image, an off-the-shelf module from IHOI [55] is used to generate the camera parameters, the hand mesh, and initial features f_0 including visual and articulation embeddings. By using the camera parameters, sampled 3D query points on the object surface are transformed into a normalized coordinate system around the hand wrist, which serve as the inputs for the subsequent structured contact codes.

Structured Contact Codes. The predicted contact states $\boldsymbol{C_v} \in \mathbb{R}^{N_v}$ are utilized to construct structured contact codes, which act as intermediate contact features. In the context of implicit reconstruction, we perform trilinear interpolation on estimated contact probabilities according to the contact point's position. In addition, to facilitate the network learning, each contact code $\boldsymbol{c_v^i} \in \mathbb{R}^1$ is mapped to a higher dimensional space by using the positional encoding [33].

Contact Code Volume. There are two disadvantages of directly extracting features from structured contact codes. First, the contact information is only limited to the mesh surface and cannot cover the surrounding space of the hand where the object is located. Second, the vertices are too sparse in 3D space to provide enough contact information as most extracted features are zero vectors. Inspired by Neural Body [40], a sparse convolutional network [15] is used to process the contact code volumes $\boldsymbol{V} = \{V_i\}_{i=1}^{L}$ at L different resolutions. Before that, the structured contact codes are scaled into the initial volume V_0. After the sparse convolution, the contact code volumes are downsampled with different spatial sizes and code dimensions, resulting in a set of contact code volumes at various resolutions. As a result, the contact code volumes are not limited to contact states at the hand mesh surface and contain diffused contact features for nearby 3D space, which is compatible with the continuous implicit functions.

Implicit Decoding. Contact code volumes of different resolutions are first normalized to the same scale [-1, 1]. Then, the contact feature fc_i is extracted by interpolation according to the query point x from each contact code volume V_i. The final contact feature fc is obtained as the concatenation of features extracted from volumes of different resolutions:

$$fc = \bigoplus(fc_1, fc_2, \cdots, fc_L) \tag{1}$$

where $\bigoplus(\cdot)$ is a concatenation operation. After that, the SDF value s on the query point x can be computed via an implicit function $\mathcal{F}$ given the conditions of the contact feature fc and the initial features f_0:

$$s = \mathcal{F}(x, fc, f_0) \tag{2}$$

Similar to other methods [9, 55], the implicit function $\mathcal{F}$ is implemented as a decoder network similar to DeepSDF [36], which composes of eight fully connected layers with a skip connection at the fourth layer.

3.3 Training Details

Contact Prediction. In the first stage, the framework is trained in an end-to-end fashion to estimate the contact region from a single image. During training, the loss $\mathcal{L}_{Contact}$ is used as follows:

$$\mathcal{L}_{Contact} = \lambda_p \mathcal{L}_{part} + \lambda_v \mathcal{L}_{vertex} + \lambda_{vs} \mathcal{L}_{vertex\_sub} \tag{3}$$

where λ_p, λ_v, and λ_{vs} are balancing weights. $\mathcal{L}_{part}$, $\mathcal{L}_{vertex}$, and $\mathcal{L}_{vertex\_sub}$ are weighted binary cross entropy (BCE) losses between the ground truth and the predicted contact probabilities. The first one corresponds to the part-level contact. For multi-scale perception and computational efficiency, the template MANO mesh is downsampled to a sub-mesh with 195 vertices for graph generation. Processed by graphormer encoders, the coarse contact prediction is generated to compute the $\mathcal{L}_{vertex\_sub}$. Then, the coarse prediction is upsampled back to 778-dimensional refined results for $\mathcal{L}_{vertex}$. Fig. 2 illustrates the contact frequency on different hand regions by analyzing statistics on a large hand-object interaction dataset OakInk [53]. It can be observed that the frequencies of different regions vary greatly. Therefore, we normalize these frequencies to $[0.1, 1]$ and use them as weight priors to compute the weighted BCE losses.

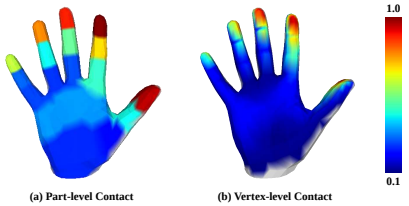

Figure 2: Visualization of contact frequency for different hand regions on OakInk [53]. (a) Part-level contact. (b) Vertex-level contact.

Implicit Reconstruction. In the second stage, the off-the-shelf module from IHOI [55] is trained together with the proposed method. Similar to IHOI, the object model is provided to guide query point sampling during training, where 95% of the points are sampled around the model surface and others uniformly in the normalized space as shown in Fig. 1. It should be noted that at test time,

query points are uniformly sampled in space since the object model is agnostic. In this part, the reconstruction loss $\mathcal{L}_{Recon}$ is calculated as follows:

$$\mathcal{L}_{Recon} = \mathcal{L}_{obj} + \mathcal{L}_{hoi} = \|s - \hat{s}\|_1 + \frac{1}{N_c}\sum_{i=1}^{N_c}(c_v^i \cdot |s_h^i|) \tag{4}$$

where $\mathcal{L}_{obj}$ is an L1 loss function between the ground truth $\hat{s}$ and the predicted SDF value s of the object similar to other approaches [8, 9, 55]. $\mathcal{L}_{hoi}$ is related to N_c vertices on the hand mesh that are in contact with the object (*i.e.*, the contact probability $c_v^i > 0.5$). s_h^i is the SDF value calculated in Equation 2 of the hand contact vertices. Taking c_v^i as the weight, $\mathcal{L}_{hoi}$ is the weighted average sum of the SDF values. This term serves as a regularization term to penalize hand contact points that penetrate or are far from the object.

4 Experiments

4.1 Implementation Details

In this work, the size of the hand-object centered image is 224×224. The number of graph nodes are $N_p = 18$ and $N_v = 778$. The length of the image feature is $D = 2048$. The shapes of contact code volumes ($L = 4$) are $V_0 = [64,\ 64,\ 64]$, $V_1 = [32,\ 32,\ 32]$, $V_2 = [16,\ 16,\ 16]$, $V_3 = [8,\ 8,\ 8]$, $V_4 = [4,\ 4,\ 4]$, and their code dimensions are $d_0 = 16$, $d_1 = 32$, $d_2 = 64$, $d_3 = d_4 = 128$. The balancing weights are $\lambda_p = 1$, $\lambda_v = \lambda_{vs} = 0.5$. The model is implemented by PyTorch [37] and the HRNet backbone [49] is pre-trained on ImageNet [49]. For both contact estimation and object reconstruction, the learning rate is set to 1e-4, and the Adam optimizer [26] is used. Each model is trained for 200 epochs on the RTX3090 GPU and the best results are reported.

4.2 Datasets and Setup

The proposed method is evaluated on two challenging real-world datasets: OakInk [53] and HO3D [18]. To our knowledge, they are two of the few benchmarks that provide official contact annotations and corresponding RGB images.

OakInk. OakInk is one of the latest and largest hand-object interaction datasets. It contains 230K images, capturing the single-hand interactions of 12 subjects with 100 objects from 32 categories. For contact prediction, the official split originally used for hand-object pose estimation [53] is adopted. The samples with a minimum distance between hand and object vertices greater than 5 mm are filtered out, resulting in 145,589 training images and 48,538 testing images. For hand-held object reconstruction, to verify the generalization of the model, we randomly select 10 objects from the above testing set and mark all their images as a new testing set. The training set only keeps images of the other 90 objects. Finally, we obtain 131,287 samples for training and 4773 for testing.

HO3D. HO3D [17] is a widely used dataset consisting of 103k images. The dataset captures 10 subjects interacting with 10 YCB objects [5]. For a fair comparison with the state-of-the-art method [55], we follow its data partition and only keep samples with contact annotations, which are provided by the latest HO3Dv3 [18] dataset.

4.3 Evaluation Metrics

For contact prediction from a single image, standard detection metrics such as precision, recall, and F1-score are adopted. For the object reconstruction, the chamfer distance (mm), F-score at 5mm and 10mm thresholds are reported. To evaluate the quality of the relation between objects and hands, the penetration depth (cm) and intersection volume (cm^3) are computed.

4.4 Experimental Results for Contact Prediction

Since there is no specific method focused on predicting hand contact regions from monocular images, we first conduct ablation experiments on model settings, then compare and validate the effectiveness of the multi-level graphormer. Finally, we evaluate different levels of contact prediction.

Table 1: Ablation study for vertex-level contact predictions on the OakInk dataset. From left to right, whether to use $\mathcal{L}_{vertex\_sub}$, whether to only predict contact (otherwise reconstruct the hand mesh at the same time), and whether to use the weighted loss.

Method	$\mathcal{L}_{vertex\_sub}$	Only Contact	Weight Prior	Precision↑	Recall↑	F1↑
M_1	✗	✗	✗	0.270	0.176	0.189
M_2	✔	✗	✗	0.285	0.196	0.210
M_3	✔	✔	✗	0.309	0.192	0.213
M_4	✔	✔	✔	**0.332**	**0.245**	**0.262**

Table 2: Comparison of different network architectures on OakInk and HO3D benchmarks.

Method	OakInk			HO3D		
	Precision↑	Recall↑	F1↑	Precision↑	Recall↑	F1↑
Only Vertex-level	0.332	**0.245**	0.262	0.476	0.422	0.416
Multi-level (Vertex output)	**0.342**	0.244	**0.262**	**0.510**	**0.441**	**0.436**
Only Part-level	0.770	0.753	0.728	0.710	0.723	0.672
Multi-level (Part output)	**0.790**	**0.767**	**0.747**	**0.722**	**0.741**	**0.685**

Ablation Study. Table 1 illustrates the quantitative ablation results for vertex-level contact predictions on the OakInk dataset. M_1 is designed to estimate the hand mesh and vertex-level contact at the same time. It yields the overall lowest detection scores. Compared with M_1, M_2 further uses the loss $\mathcal{L}_{vertex\_sub}$ calculated on the sub-mesh proposed in Section 3.3. The precision, recall, and F1-score are improved by 5.6%, 11.4%, and 11.1%, respectively, proving the effectiveness of multi-scale features aggregation based on the hand model in this task. Different from M_2, M_3 does not reconstruct the hand mesh and only performs hand contact prediction. Although the recall drops slightly, its precision improves by 8.4%, showing that focusing on a single task could make the network learn more effectively. Finally, compared with M_3, M_4 uses weight priors for BCE losses in Equation 2 and achieves a huge boost on all metrics (*e.g.*, 27.6% on recall and 23.0% on F1), showing that the weight priors of contacts introduced in Section 3.3 can provide useful guidance for the model.

Effectiveness of Multi-level Graphormer. In this work, three network architectures are trained and evaluated on OakInk and HO3D benchmarks, respectively. As shown in Table 2, in addition to the multi-level graphormer encoders, we also use the single-level model in Fig. 1 for contact prediction. The outputs of the multi-level method are compared with corresponding single-level outputs. Though we observe that a single vertex-level model yields slightly better recall on OakInk, the multi-level one can improve the precision from 0.332 to 0.342 benefiting from using the part-level output to refine features for vertices. Regarding the part-level output, the coarse-to-fine model outperforms the single part-level model on all three evaluation metrics. The multi-level model also achieves superior performance on HO3D for all three metrics, which further demonstrates the advantage of using the proposed coarse-to-fine learning framework. Fig. 3 further illustrates the qualitative results of the proposed method on two benchmarks. Benefiting from accumulating both global contexts and local details by using the graph-based transformer, the proposed method is robust to input images with hand or object occlusions.

Part-level vs. Vertex-level Prediction. In Table 2, the vertex-level predictions are worse than the part-level results, showing that the dense vertex-level prediction is more difficult than the sparse one. For the single-level architecture, the F1 score of the vertex-level method on OakInk is only 0.262, while the single part-level model achieves 0.728. On the HO3D dataset, we can observe a similar performance gap between the part-level and vertex-level accuracy. Fig. 3 illustrates that part-level predictions are closer to the ground truth than vertex-level predictions. Therefore, part-level predictions are propagated to hand mesh vertices and are used in subsequent experiments. Please see the supplementary material for more details and analysis.

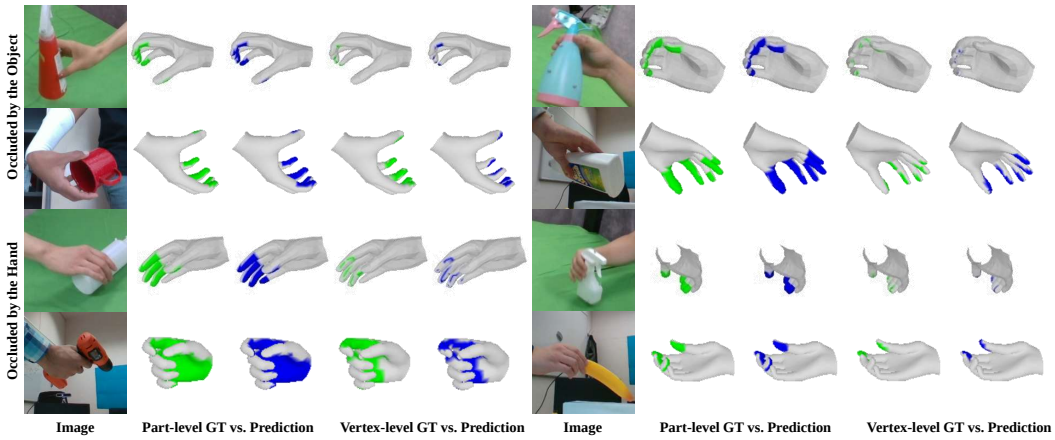

Figure 3: Visualizations of contact prediction on OakInk (Rows 1, 3) and HO3D (Rows 2, 4) datasets. Since the method only estimates contact, the result is rendered on the ground truth hand mesh. For samples whose contact regions are occluded by hands, hand meshes are rotated 180 degrees for clear visualization. The proposed method is robust to both hand and object occlusions.

Table 3: Comparison with the state-of-the-art method on the HO3D benchmark on F-score at 5mm and 10mm thresholds, chamfer distance (CD), penetration depth (PD), and intersection volume (IV). Two types of hand meshes from the off-the-shelf module and ground truth are considered separately.

Method	Hand Mesh	F@5mm↑	F@10mm↑	CD (mm)↓	PD (cm)↓	IV (cm^3)↓
IHOI [55]	Prediction	0.326	0.566	0.835	1.13	**5.14**
Ours		**0.338**	**0.580**	**0.807**	**0.93**	5.18
IHOI [55]	Ground Truth	0.351	0.600	0.656	0.90	4.10
Ours		**0.393**	**0.633**	**0.646**	**0.67**	**2.91**

Table 4: Comparison with the state-of-the-art method on the OakInk benchmark. The HOI loss $\mathcal{L}_{hoi}$ is further validated as a variable.

Method	F@5mm↑	F@10mm↑	CD (mm)↓	PD (cm)↓	IV (cm^3)↓
IHOI [55]	0.432	0.658	0.491	0.75	4.36
Ours w/o $\mathcal{L}_{hoi}$	0.447	0.716	0.274	0.66	3.03
Ours with $\mathcal{L}_{hoi}$	**0.459**	**0.718**	**0.260**	**0.62**	**2.67**

4.5 Experimental Results on Hand-held Object Reconstruction

In this section, we use the recent state-of-the-art IHOI [55] as a strong baseline for our method. We present quantitative and qualitative comparisons on challenging HO3D and OakInk benchmarks.

Quantitative comparison on the HO3D Dataset. Since this task only focuses on reconstructing objects without hands, we follow the setting of IHOI [55] and use either predicted hand meshes from [46] or ground-truth hand meshes to benefit the 3D reconstruction of hand-held objects. As shown in Table 3, when our model uses predicted hand meshes, we observe that our method can improve $\mathrm{F@5mm}$ and $\mathrm{F@10mm}$ by 3.7% and 2.5% and greatly reduce the chamfer distance. Although the intersection volume is slightly larger by 0.04 cm^3, our method achieves a 17.7% improvement in terms of the penetration depth. When the hand mesh is perfect, our method shows a more obvious advantage and consistently outperforms IHOI across all metrics. It largely improves $\mathrm{F@5mm}$ by 12.0% and reduces the penetration depth by 23.3%.

Quantitative Comparison on the OakInk Benchmark. To show that our model can work well for unseen objects, we split the dataset to make sure that testing objects do not exist in the training set. As shown in Table 4, when our model is not trained together with L_{hoi}, it can still outperform

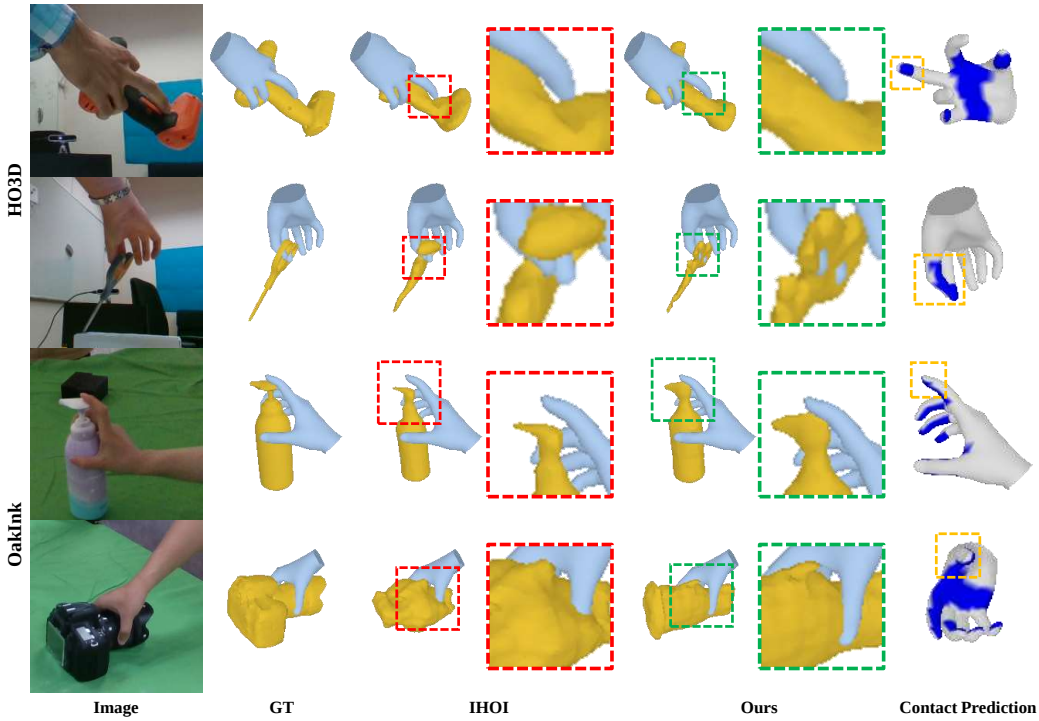

Figure 4: Qualitative comparison with the state-of-the-art method on the HO3D and OakInk datasets. Our method can reconstruct more realistic objects, especially for parts that are in contact with hands.

IHOI on all metrics. Our final model, which is learned with L_{hoi}, achieves even better results on different metrics. Compared with IHOI, our method can largely improve $\mathrm{F@5mm}$ and $\mathrm{F@10mm}$ by 6.3% and 9.1%, respectively. At the same time, it reduces the penetration depth and intersection volume by 17.3% and 38.8%, which suggests our model can reconstruct more realistic objects that naturally interact with hands.

Qualitative Comparison. Fig. 4 illustrates qualitative comparisons on the HO3D and OakInk datasets. Compared with the state-of-the-art IHOI [55] (red dotted box), our method shows a clear advantage in the reconstruction of object parts that are in contact with the hand. It can be seen that the predicted hand contacts (yellow dashed box) provide effective guidance to recover corresponding object parts (green dashed box). We also observe that our method is robust to occlusions. As illustrated in the first and fourth rows in Fig. 4, our model can still work well when objects are occluded by hands. For unseen objects with complex structures (*e.g.*, camera) in OakInk, our model can also obtain realistic results. More qualitative results can be found in the supplementary material.

5 Conclusion and Discussion

Conclusion. This paper introduces a novel representation of explicit contacts for the implicit reconstruction of hand-held objects. First, the multi-level graph-based transformer encoders are cascaded to estimate accurate 3D hand-object contacts from a single RGB image. Then, the predicted contact states are anchored to the hand surface and diffused to the nearby space to construct the implicit neural representation for the manipulated object. Extensive experiments on HO3D and OakInk datasets indicate that our method can pay more attention to the object parts that are in contact with hands and reconstruct more realistic object meshes.

Limitations. The proposed method currently focuses on hand-held object reconstruction. Although the learned contact states could be used to effectively improve the object shape reconstruction, it relies on good hand reconstruction. In addition, it is still challenging to reconstruct details of unseen objects. In future work, we attempt to integrate the hand reconstruction module for better hand-object interaction reconstruction and leverage object category priors to improve generalization.

References

[1] Seungryul Baek, Kwang In Kim, and Tae-Kyun Kim. Pushing the envelope for RGB-based dense 3D hand pose estimation via neural rendering. In *CVPR*, 2019.

[2] Antonio Bicchi and Vijay Kumar. Robotic grasping and contact: A review. In *ICRA*, 2000.

[3] Adnane Boukhayma, Rodrigo de Bem, and Philip HS Torr. 3D hand shape and pose from images in the wild. In *CVPR*, 2019.

[4] Gereon Buescher, Martin Meier, Guillaume Walck, Robert Haschke, and Helge J Ritter. Augmenting curved robot surfaces with soft tactile skin. In *IROS*, 2015.

[5] Berk Calli, Arjun Singh, Aaron Walsman, Siddhartha Srinivasa, Pieter Abbeel, and Aaron M Dollar. The ycb object and model set: Towards common benchmarks for manipulation research. In *ICAR*, 2015.

[6] Yu-Wei Chao, Wei Yang, Yu Xiang, Pavlo Molchanov, Ankur Handa, Jonathan Tremblay, Yashraj S Narang, Karl Van Wyk, Umar Iqbal, Stan Birchfield, et al. DexYCB: A benchmark for capturing hand grasping of objects. In *CVPR*, 2021.

[7] Xingyu Chen, Yufeng Liu, Chongyang Ma, Jianlong Chang, Huayan Wang, Tian Chen, Xiaoyan Guo, Pengfei Wan, and Wen Zheng. Camera-space hand mesh recovery via semantic aggregation and adaptive 2D-1D registration. In *CVPR*, 2021.

[8] Zerui Chen, Shizhe Chen, Cordelia Schmid, and Ivan Laptev. gSDF: Geometry-Driven signed distance functions for 3D hand-object reconstruction. In *CVPR*, 2023.

[9] Zerui Chen, Yana Hasson, Cordelia Schmid, and Ivan Laptev. AlignSDF: Pose-aligned signed distance fields for hand-object reconstruction. In *ECCV*, 2022.

[10] Zhiqin Chen and Hao Zhang. Learning implicit fields for generative shape modeling. In *CVPR*, 2019.

[11] Christopher B Choy, Danfei Xu, JunYoung Gwak, Kevin Chen, and Silvio Savarese. 3D-R2N2: A unified approach for single and multi-view 3D object reconstruction. In *ECCV*, 2016.

[12] Mihai Fieraru, Mihai Zanfir, Elisabeta Oneata, Alin-Ionut Popa, Vlad Olaru, and Cristian Sminchisescu. Three-dimensional reconstruction of human interactions. In *CVPR*, 2020.

[13] Mihai Fieraru, Mihai Zanfir, Elisabeta Oneata, Alin-Ionut Popa, Vlad Olaru, and Cristian Sminchisescu. Learning complex 3d human self-contact. In *AAAI*, 2021.

[14] Patrick Grady, Chengcheng Tang, Christopher D Twigg, Minh Vo, Samarth Brahmbhatt, and Charles C Kemp. ContactOpt: Optimizing contact to improve grasps. In *CVPR*, 2021.

[15] Benjamin Graham, Martin Engelcke, and Laurens Van Der Maaten. 3D semantic segmentation with submanifold sparse convolutional networks. In *CVPR*, 2018.

[16] Thibault Groueix, Matthew Fisher, Vladimir G Kim, Bryan C Russell, and Mathieu Aubry. A papier-mâché approach to learning 3d surface generation. In *CVPR*, 2018.

[17] Shreyas Hampali, Mahdi Rad, Markus Oberweger, and Vincent Lepetit. HOnnotate: A method for 3D annotation of hand and object poses. In *CVPR*, 2020.

[18] Shreyas Hampali, Sayan Deb Sarkar, and Vincent Lepetit. Ho-3d_v3: Improving the accuracy of hand-object annotations of the HO-3D dataset. *arXiv preprint arXiv:2107.00887*, 2021.

[19] Shreyas Hampali, Sayan Deb Sarkar, Mahdi Rad, and Vincent Lepetit. Keypoint Transformer: Solving joint identification in challenging hands and object interactions for accurate 3D pose estimation. In *CVPR*, 2022.

[20] Yana Hasson, Bugra Tekin, Federica Bogo, Ivan Laptev, Marc Pollefeys, and Cordelia Schmid. Leveraging photometric consistency over time for sparsely supervised hand-object reconstruction. In *CVPR*, 2020.

[21] Yana Hasson, Gul Varol, Dimitrios Tzionas, Igor Kalevatykh, Michael J Black, Ivan Laptev, and Cordelia Schmid. Learning joint reconstruction of hands and manipulated objects. In *CVPR*, 2019.

[22] Haoyu Hu, Xinyu Yi, Hao Zhang, Jun-Hai Yong, and Feng Xu. Physical interaction: Reconstructing hand-object interactions with physics. In *SIGGRAPH Asia*, 2022.

[23] Chun-Hao P Huang, Hongwei Yi, Markus Höschle, Matvey Safroshkin, Tsvetelina Alexiadis, Senya Polikovsky, Daniel Scharstein, and Michael J Black. Capturing and inferring dense full-body human-scene contact. In *CVPR*, 2022.

[24] Divye Jain, Andrew Li, Shivam Singhal, Aravind Rajeswaran, Vikash Kumar, and Emanuel Todorov. Learning deep visuomotor policies for dexterous hand manipulation. In *ICRA*, 2019.

[25] Korrawe Karunratanakul, Jinlong Yang, Yan Zhang, Michael J Black, Krikamol Muandet, and Siyu Tang. Grasping Field: Learning implicit representations for human grasps. In *3DV*, 2020.

[26] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *ICLR*, 2015.

[27] Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. In *ICLR*, 2017.

[28] Dominik Kulon, Riza Alp Güler, I. Kokkinos, M. Bronstein, and S. Zafeiriou. Weakly-supervised mesh-convolutional hand reconstruction in the wild. In *CVPR*, 2020.

[29] Nikolaos Kyriazis and Antonis Argyros. Scalable 3D tracking of multiple interacting objects. In *CVPR*, 2014.

[30] Qiang Li, Oliver Kroemer, Zhe Su, Filipe Fernandes Veiga, Mohsen Kaboli, and Helge Joachim Ritter. A review of tactile information: Perception and action through touch. *TRO*, 2020.

[31] Kevin Lin, Lijuan Wang, and Zicheng Liu. Mesh graphormer. In *CVPR*, 2021.

[32] Lars Mescheder, Michael Oechsle, Michael Niemeyer, Sebastian Nowozin, and Andreas Geiger. Occupancy Networks: Learning 3D reconstruction in function space. In *CVPR*, 2019.

[33] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. NeRF: Representing scenes as neural radiance fields for view synthesis. In *ECCV*, 2022.

[34] Joseph L. Mundy. Object recognition in the geometric era: A retrospective. In *Toward Category-Level Object Recognition*, Lecture Notes in Computer Science, 2006.

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

[36] Jeong Joon Park, Peter Florence, Julian Straub, Richard Newcombe, and Steven Lovegrove. DeepSDF: Learning continuous signed distance functions for shape representation. In *CVPR*, 2019.

[37] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *NeurIPS*, 2019.

[38] Georgios Pavlakos, Xiaowei Zhou, Konstantinos G Derpanis, and Kostas Daniilidis. Coarse-to-fine volumetric prediction for single-image 3D human pose. In *CVPR*, 2017.

[39] Songyou Peng, Chiyu Jiang, Yiyi Liao, Michael Niemeyer, Marc Pollefeys, and Andreas Geiger. Shape as points: A differentiable poisson solver. *NeurIPS*, 2021.

[40] Sida Peng, Yuanqing Zhang, Yinghao Xu, Qianqian Wang, Qing Shuai, Hujun Bao, and Xiaowei Zhou. Neural body: Implicit neural representations with structured latent codes for novel view synthesis of dynamic humans. In *CVPR*, 2021.

[41] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. PointNet: Deep learning on point sets for 3D classification and segmentation. In *CVPR*, 2017.

[42] Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J Guibas. PointNet++: Deep hierarchical feature learning on point sets in a metric space. *NeurIPS*, 2017.

[43] Gernot Riegler, Ali Osman Ulusoy, and Andreas Geiger. OctNet: Learning deep 3D representations at high resolutions. In *CVPR*, 2017.

[44] Lawrence Gilman Roberts. *Machine perception of three-dimensional solids*. PhD thesis, Massachusetts Institute of Technology, 1963.

[45] Javier Romero, Dimitrios Tzionas, and Michael J. Black. Embodied Hands: Modeling and capturing hands and bodies together. *TOG*, 2017.

[46] Yu Rong, Takaaki Shiratori, and Hanbyul Joo. FrankMocap: Fast monocular 3D hand and body motion capture by regression and integration. *arXiv preprint arXiv:2008.08324*, 2020.

[47] Tze Ho Elden Tse, Kwang In Kim, Ales Leonardis, and Hyung Jin Chang. Collaborative learning for hand and object reconstruction with attention-guided graph convolution. In *CVPR*, 2022.

[48] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NeurIPS*, volume 30, 2017.

[49] Jingdong Wang, Ke Sun, Tianheng Cheng, Borui Jiang, Chaorui Deng, Yang Zhao, Dong Liu, Yadong Mu, Mingkui Tan, Xinggang Wang, et al. Deep high-resolution representation learning for visual recognition. *TPAMI*, 2020.

[50] Nanyang Wang, Yinda Zhang, Zhuwen Li, Yanwei Fu, Wei Liu, and Yu-Gang Jiang. Pixel2mesh: Generating 3D mesh models from single RGB images. In *ECCV*, 2018.

[51] 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.

[52] Lixin Yang, Kailin Li, Xinyu Zhan, Jun Lv, Wenqiang Xu, Jiefeng Li, and Cewu Lu. ArtiBoost: Boosting articulated 3D hand-object pose estimation via online exploration and synthesis. In *CVPR*, 2022.

[53] 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.

[54] Lixin Yang, Xinyu Zhan, Kailin Li, Wenqiang Xu, Jiefeng Li, and Cewu Lu. CPF: Learning a contact potential field to model the hand-object interaction. In *ICCV*, 2021.

[55] Yufei Ye, Abhinav Gupta, and Shubham Tulsiani. What's in your hands? 3D reconstruction of generic objects in hands. In *CVPR*, 2022.

[56] Zhao-Heng Yin, Binghao Huang, Yuzhe Qin, Qifeng Chen, and Xiaolong Wang. Rotating without seeing: Towards in-hand dexterity through touch. *RSS*, 2023.

[57] Hao Zhang, Yuxiao Zhou, Yifei Tian, Jun-Hai Yong, and Feng Xu. Single depth view based real-time reconstruction of hand-object interactions. *TOG*, 2021.

[58] Zimeng Zhao, Binghui Zuo, Wei Xie, and Yangang Wang. Stability-driven contact reconstruction from monocular color images. In *CVPR*, 2022.

Appendix

This supplementary material provides additional details of the proposed method and more experimental results. In Section A, we present more details of our network architecture. Section B presents more details about the datasets that we use in our experiments. In Section C, we conduct ablation experiments about using contact states. Finally, we include more qualitative results in Section D. We qualitatively compare our method against the state of the art [55] in *results_video.mp4*. Our code will be publicly available.

A Network Architecture

In Fig. 5, we illustrate more details about our network architecture. The proposed graph-based transformer consists of graphormer blocks proposed in [31]. For the part-level graph-based transformer (Fig. 5 (a)), one block is sufficient while three blocks are utilized for the vertex-level estimator (Fig. 5 (b)). The intermediate features and their dimensions can be found in the figure. In addition, the computation of the hand-object interaction loss $\mathcal{L}_{hoi}$ is shown in Fig. 5 (c). The dimension of the extracted contact feature is 352, and the initial features F [55] including visual and articulation embeddings are 304 dimensions. A linear layer is used to downsample the concatenated features to 304 dimensions for a fair comparison with IHOI [55].

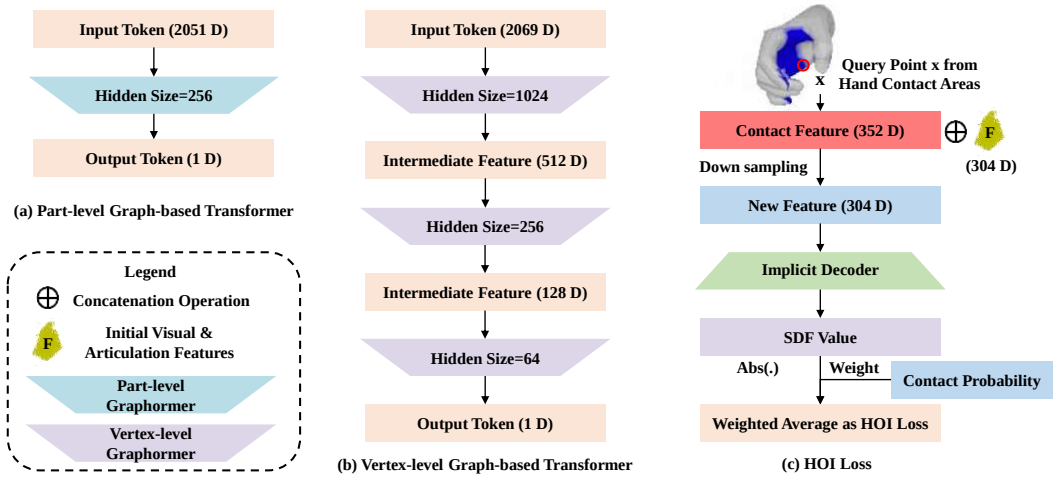

Figure 5: Network Architecture. (a) The architecture of the part-level graph-based transformer. (b) The architecture of the vertex-level graph-based transformer. (c) The computation process of $\mathcal{L}_{hoi}$.

B Dataset Details

In this paper, two real-world datasets, OakInk [53] and HO3D [18] are used for comparison. The OakInk benchmark contains 100 objects from 32 categories. For the hand-held object reconstruction, the official split originally used for hand-object pose estimation [53] is further divided by different objects to verify the generalization of the model. Specifically, 10 objects (i.e., A01027, A02031, C12001, C35001, C91001, O50001, S10008, S10018, S16003, Y35037) are used for testing and the remaining 90 are used for training. HO3D [17] consists of 10 subjects interacting with 10 YCB objects [5]. Following the data partition of IHOI [55] and retrieving samples with contact annotations, we obtain 64,775 images for training and 1032 images for testing.

C Ablation Study on the Contact States

In this work, the predicted contact states $\boldsymbol{C_v} \in \mathbb{R}^{N_v}$ are utilized to build structured contact codes for the implicit reconstruction. As mentioned in Line 282 of the main text, current vertex-level predictions are far less accurate than part-level ones, so the part-level contact probability $\boldsymbol{C_p} \in \mathbb{R}^{N_p}$ generated in the first stage is adopted. More comparisons and analyses of the two datasets are provided as follows.

Table 5: Comparison of different contact states on the HO3D benchmark on F-score at 5mm and 10mm thresholds, chamfer distance (CD), penetration depth (PD), and intersection volume (IV). Two types of states from the contact prediction and ground truth are considered separately. The baseline IHOI is also provided for comparison.

Contact Source	States	F@5mm↑	F@10mm↑	CD (mm)↓	PD (cm)↓	IV (cm^3)↓
Prediction	Part	**0.393**	**0.633**	**0.646**	**0.67**	**2.91**
	Vertex	0.374	0.612	0.741	0.78	4.05
Ground Truth	Part	**0.490**	0.699	0.655	0.50	2.15
	Vertex	0.488	**0.709**	**0.583**	**0.49**	**1.90**
IHOI [55]	-	0.351	0.600	0.656	0.90	4.10

Table 6: Comparison of different contact states on the OakInk benchmark.

Contact Source	States	F@5mm↑	F@10mm↑	CD (mm)↓	PD (cm)↓	IV (cm^3)↓
Prediction	Part	0.447	**0.716**	**0.274**	0.66	3.03
	Vertex	**0.450**	0.704	0.288	**0.60**	**2.70**
Ground Truth	Part	**0.461**	**0.720**	0.281	0.62	2.73
	Vertex	0.446	0.711	**0.280**	**0.55**	**2.41**
IHOI [55]	-	0.432	0.658	0.491	0.75	4.36

Ablation on the HO3D Dataset. In Table 5, the first two rows show quantitative results for object reconstruction using part-level and vertex-level contacts estimated in the first stage, respectively. We observe that both of them show significant improvement over IHOI [55] which does not take the hand-object contact into account, and the former outperforms the latter. Compared with the method using vertex-level contact prediction, the part-level one could increase $\mathrm{F@5mm}$ and $\mathrm{F@10mm}$ by 5.1% and 3.4% and greatly reduce the chamfer distance. In addition, the intersection volume (2.91 cm^3) of the part-level state achieves a 28.1% improvement over the other one (4.05 cm^3). These results can be explained by the fact that part-level predictions are more accurate. When the contact is perfect (Rows 3-4), our method shows a more obvious advantage over IHOI across all metrics. For example, the part-level state largely improves $\mathrm{F@5mm}$ by 39.6% and reduces the penetration depth by 44.4% than IHOI. What's more, though the $\mathrm{F@5mm}$ (0.490) is 0.002 larger than that of the vertex level (0.488), the other metrics show a clear advantage of the vertex-level model against the part-level model, which also indicates that precise vertex-level contacts can lead to better results.

Ablation on the OakInk Dataset. As illustrated in Table 6, all of our methods (the first four rows) outperform the state-of-the-art method [55] on all metrics. For models using the predicted contacts, compared with the vertex-level model, the part-level model achieves a 1.7% improvement for $\mathrm{F@10mm}$ and a 4.9% improvement for chamfer distance. However, the penetration depth and the intersection volume of the part level are slightly worse than the vertex level, indicating that the vertex-level contacts are more efficient for interaction reconstruction. Furthermore, Table 6 shows that compared with using predicted contacts, the ground truth contacts only bring limited improvement, such as increasing the $\mathrm{F@5mm}$ of the part-level model from 0.447 to 0.461 and decreasing the penetration depth of the vertex-level model from 0.60 cm to 0.55 cm. Similar to the conclusion in Table 5, the ground truth vertex-level contacts could lead to the best intersection volume (2.41 cm^3). However, its object reconstruction is not as good as the model using estimated vertex-level predictions. The main reason is that ground-truth contact annotations on OakInk are not accurate, which also brings negative impacts to the final reconstruction results. Therefore, it is reasonable to use more accurate part-level contact predictions for the implicit object reconstruction when precise vertex-level contact annotations are not available.

D Qualitative Results.

Contact Prediction. Fig. 6 and Fig. 7 show the qualitative results of predicting contact regions from a single RGB image on two datasets. It can be observed that the proposed method is robust to complex hand poses and occlusions from hands or objects. Our approach can produce reasonable results even for challenging samples such as picking up a drill or using scissors. In addition, the contact regions predicted using a single module are shown in red. Compared with the multi-level structure, the single-level method cannot accurately predict the contact area when dealing with occluded samples, which indicates the superiority of the multi-level architecture.

Object Reconstruction. Fig. 8 shows more qualitative results with the state-of-the-art method [55]. As shown in the results of the first five rows, for unseen objects during training (*i.e.*, from the OakInk dataset), the proposed method can still perform well, especially for object parts that are in contact with the hand. In addition, since the structured contact codes provide both positional and contact information, our approach can still produce plausible results for occluded object parts.

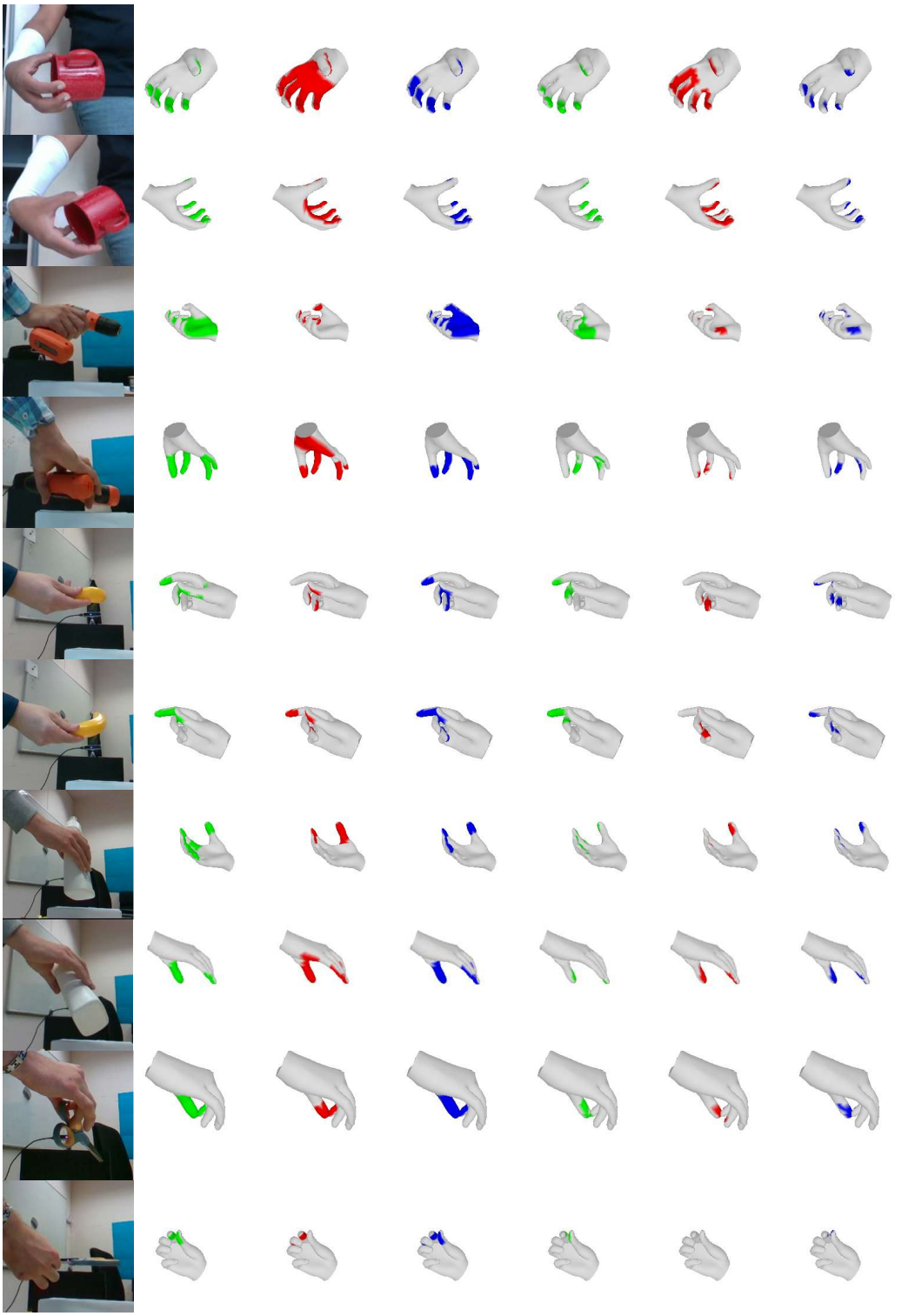

Figure 6: Contact prediction on the HO3D dataset. The ground truth area is green, the single part or vertex-level prediction is red, and the multi-level estimation corresponds to blue. For the sample whose contact regions are occluded by hands, the hand mesh is rotated 90 or 180 degrees for clear visualization.

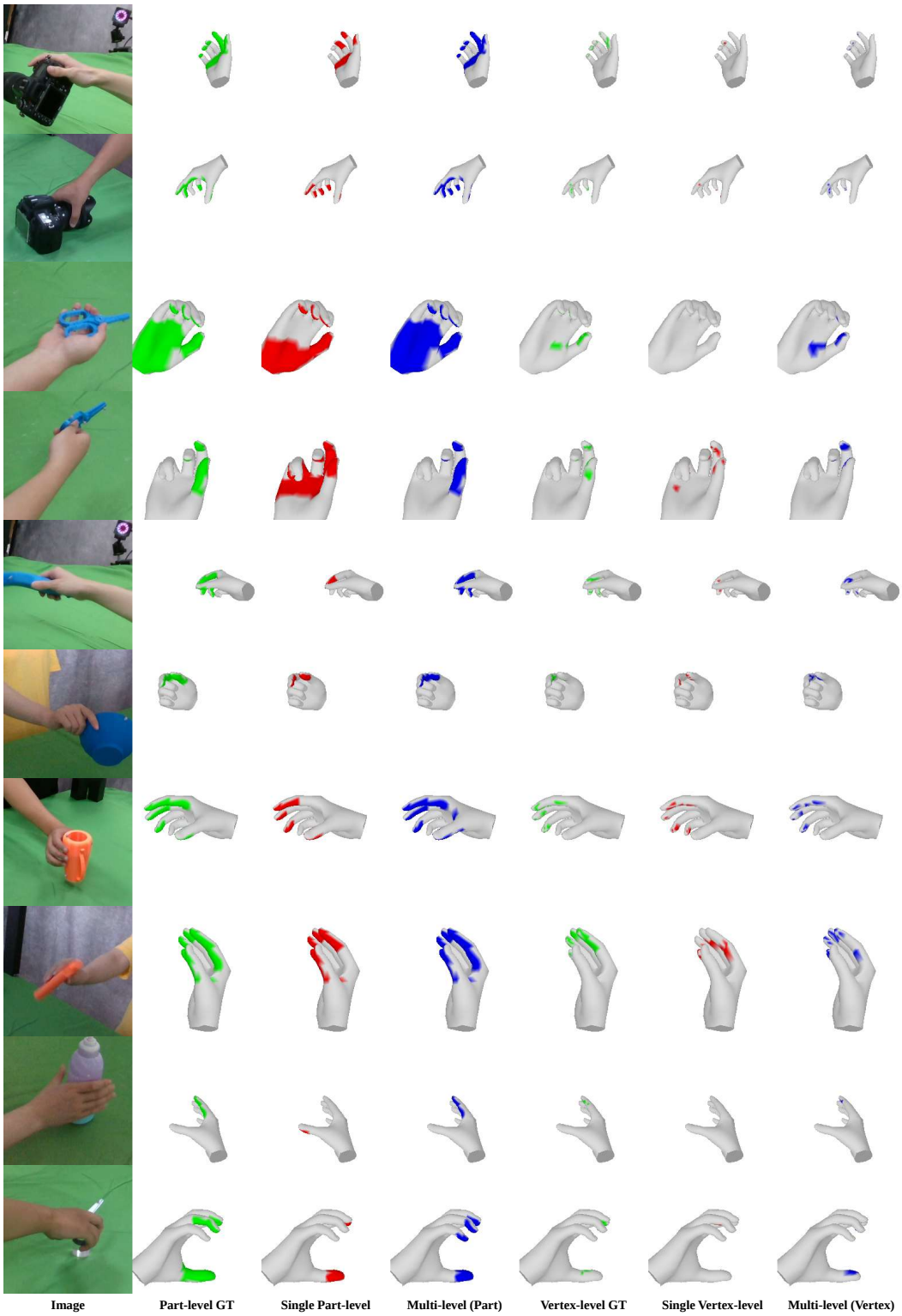

Figure 7: Contact prediction on the OakInk dataset. The ground truth area is green, the single part or vertex-level prediction is red, and the multi-level estimation corresponds to blue. For the sample whose contact regions are occluded by hands, the hand mesh is rotated 90 or 180 degrees for clear visualization.

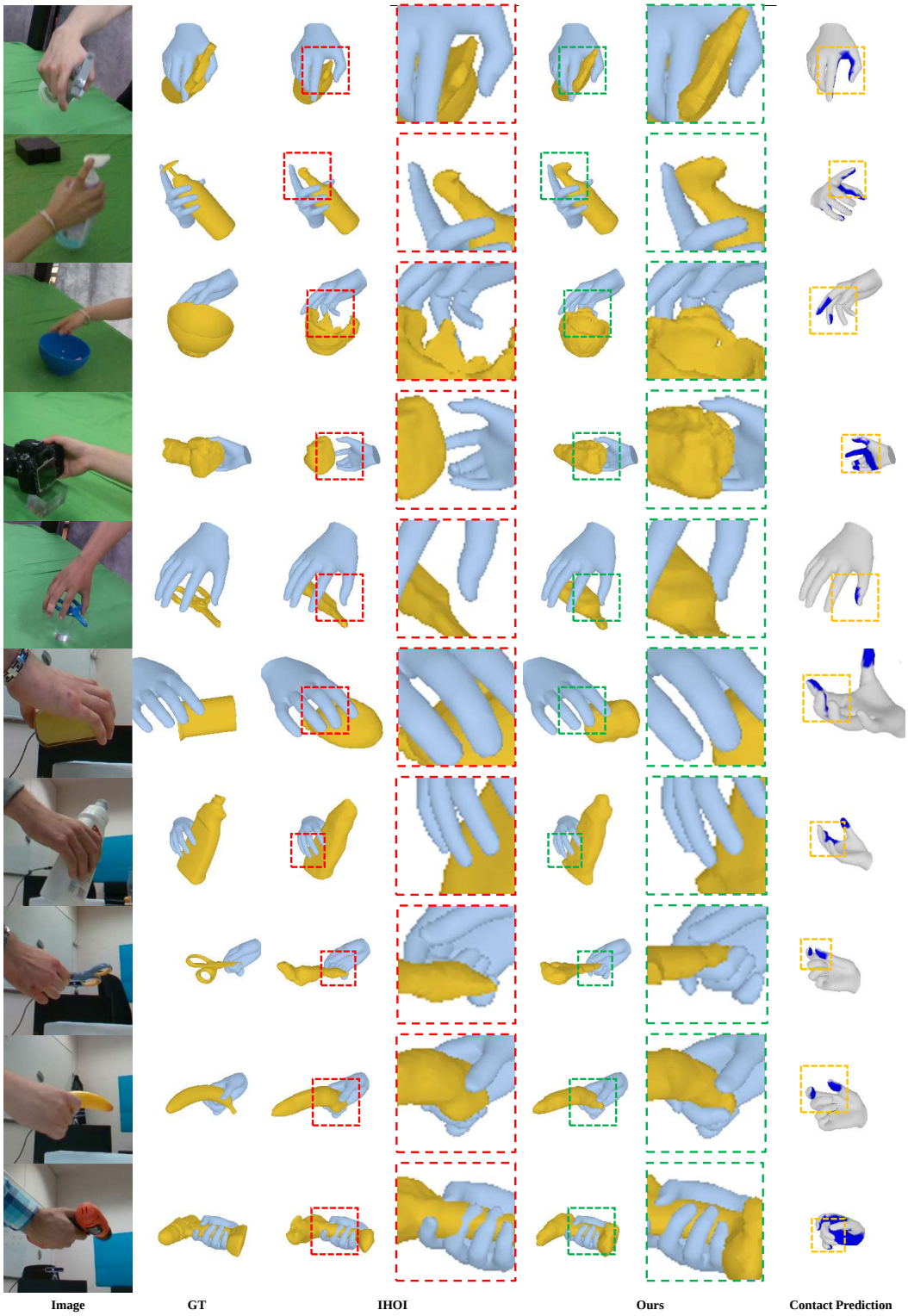

Figure 8: Qualitative comparison with the state-of-the-art method on the OakInk (Rows 1-5) and HO3D (Rows 6-10) datasets. The proposed method is more robust to occlusions and unseen objects during training than IHOI.